

PENERAPAN METODE AGILE SCRUM DALAM PENGEMBANGAN SISTEM INFORMASI ADMINISTRASI KEGIATAN COASS (CO-ASSISTANT) PADA RUMAH SAKIT PENDIDIKAN GIGI DAN MULUT

I Gusti Ngurah Ari Wira Nugraha¹, Ngurah Indra ER², Komang Sri Utami², Fajar Purnama², I Wayan Shandyasa²

¹Mahasiswa Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana

²Dosen Program Studi Teknik Elektro, Fakultas Teknik, Universitas Udayana

Jalan Raya Kampus Unud, Jimbaran, Bali

wira.nugraha133@student.unud.ac.id, indra@unud.ac.id, sriutami@unud.ac.id,

fajarpurnama@unud.ac.id, shandyasa@unud.ac.id

ABSTRAK

Administrasi program Co-Assistant (Coass) di Rumah Sakit Pendidikan Gigi dan Mulut masih dilakukan secara manual, yang berdampak pada pencatatan data yang lambat, kurang terstruktur, dan minim transparansi. Penelitian ini bertujuan mengeksplorasi penerapan metode Agile Scrum dalam pengembangan sistem informasi berbasis web untuk mendukung proses administrasi tersebut, serta mengevaluasi sejauh mana pendekatan ini sesuai dan bermanfaat dalam konteks lingkungan rumah sakit pendidikan yang dinamis. Pengembangan dilakukan dalam lima *sprint* iteratif dengan total 106 *story point*. Evaluasi dilakukan secara deskriptif berdasarkan penyelesaian *backlog*, pengalaman tim dalam mengelola kompleksitas logika bisnis, serta sejauh mana fitur yang dikembangkan mencerminkan kebutuhan pengguna. Hasilnya menunjukkan bahwa metode Scrum mendukung proses pengembangan yang terstruktur dan fleksibel, serta memudahkan tim dalam menangani tantangan integrasi antar modul. Penyesuaian sistem terhadap alur kerja pengguna akhir, seperti manajemen *logbook* dan mekanisme penilaian klinis, menunjukkan bahwa pendekatan ini berpotensi diterapkan pada sistem informasi serupa. Penelitian ini memberikan kontribusi awal berupa model pengembangan adaptif yang dapat dijadikan acuan bagi institusi pendidikan kesehatan dalam mempercepat digitalisasi proses administrasi akademis-klinis.

Kata kunci : Agile Development, Scrum, Sistem Informasi, Pengembangan Perangkat Lunak, Coass, Rumah Sakit Pendidikan Gigi dan Mulut

ABSTRACT

The manual administration of the Co-Assistant (Coass) program at Dental and Oral Education Hospitals is often inefficient, prone to delays, and lacks transparency in data handling. To address these limitations, this study implemented a web-based information system developed using the Agile Scrum methodology. Development was carried out through five iterative sprints, totaling 106 story points, enabling the team to incrementally build and refine system features in alignment with stakeholder needs. The Scrum approach not only facilitated the completion of the product backlog but also proved instrumental in managing complex business logic and ensuring seamless inter-module integration. Key outcomes include enhanced efficiency in logbook recording, streamlined clinical assessment workflows, and improved data transparency. These results suggest that Agile Scrum is well-suited for dynamic, semi-academic environments such as teaching hospitals. This research offers a replicable, adaptive development model that can guide other health education institutions in accelerating the digital transformation of their academic and clinical administrative processes.

Key Words : Agile Development, Scrum, Information System, Coass Administration, Software Development Dental and Oral Education Hospital

1. PENDAHULUAN

Pendidikan profesi dokter gigi merupakan bagian penting dalam sistem pelayanan kesehatan yang bertujuan menghasilkan tenaga medis yang kompeten dan profesional. Salah satu tahapan krusial dalam pendidikan ini adalah program *Co-Assistant (Coass)*, di mana mahasiswa dituntut untuk mempraktikkan ilmu kedokteran gigi secara langsung kepada pasien guna memenuhi kompetensi utama sebagai seorang dokter dengan mempraktikkan ilmu yang telah dipelajari secara langsung pada pasien [1]. Sesuai dengan ketentuan Konsil Kedokteran Indonesia (KKI), dokter gigi harus menunjukkan sikap profesional dalam memberikan pelayanan kesehatan [2], dan hal ini dapat ditingkatkan melalui pengalaman klinis yang disediakan oleh rumah sakit pendidikan [3].

Sebagai pusat pelayanan sekaligus pendidikan, Rumah Sakit Gigi dan Mulut (RSGM) memiliki tanggung jawab dalam memfasilitasi dan mengevaluasi kompetensi mahasiswa Coass. Aktivitas administrasi Coass di RSGM meliputi pendaftaran mahasiswa, pengelolaan *requirement stage*, pencatatan *logbook*, hingga evaluasi dan penilaian. Namun, saat ini masih ada beberapa RSGM yang sebagian besar proses administrasinya masih dilakukan secara manual dengan dokumen fisik atau *spreadsheet* sederhana. Ketiadaan sistem informasi yang terintegrasi menyebabkan proses supervisi dan evaluasi pencapaian kompetensi menjadi kurang efektif. Penggunaan dokumen fisik dalam pencatatan *logbook* Coass dinilai tidak efisien karena memerlukan waktu yang panjang, rentan terhadap kehilangan atau kerusakan data, serta menyulitkan proses pemantauan oleh pihak terkait [4]. Di sisi lain, pengolahan data penilaian menggunakan Microsoft Excel juga dianggap kurang optimal, karena berpotensi menimbulkan duplikasi, rawan kesalahan

memasukkan data, dan tidak mendukung efisiensi dalam pembuatan laporan [5].

Mengatasi permasalahan tersebut, teknologi informasi menjadi solusi yang relevan dan dibutuhkan. Sistem informasi berbasis web yang dirancang khusus untuk pengelolaan administrasi Coass dapat meningkatkan efisiensi dan akurasi data, memudahkan koordinasi antara pihak fakultas, rumah sakit, dan mahasiswa, serta mendukung evaluasi kompetensi mahasiswa secara lebih objektif dan *real-time* [6].

Pengembangan sistem informasi dalam konteks pendidikan klinis memiliki tantangan tersendiri, yaitu kebutuhan pengguna yang dapat berubah seiring waktu. Metode pengembangan perangkat lunak tradisional seperti *Waterfall* kurang mampu menyesuaikan diri dengan perubahan kebutuhan pengguna yang sering kali terjadi di tengah proses. Oleh karena itu, pendekatan *Agile Development* dengan kerangka kerja *Scrum* dipilih dalam penelitian ini. *Scrum* merupakan metode iteratif yang menekankan kolaborasi tim dan adaptasi berkelanjutan terhadap perubahan kebutuhan [7]. Kerangka kerja ini mengorganisasi proses pengembangan dalam *sprint* yang terukur dan disertai evaluasi periodik seperti *Sprint Review* dan *Sprint Retrospective*, sehingga memfasilitasi perbaikan terus-menerus [8].

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk menganalisis dan melaporkan penerapan metode *Agile Development* model *Scrum* dalam proses pengembangan sistem informasi administrasi kegiatan Coass pada RSGM. Fokus utama terletak pada tahapan perencanaan, pelaksanaan, hingga evaluasi *sprint*, serta efektivitas metode *Scrum* dalam menyelesaikan *backlog* dan menyesuaikan dengan kebutuhan pengguna selama proses pengembangan berlangsung.

2. Kajian Pustaka

2.1 Agile Development

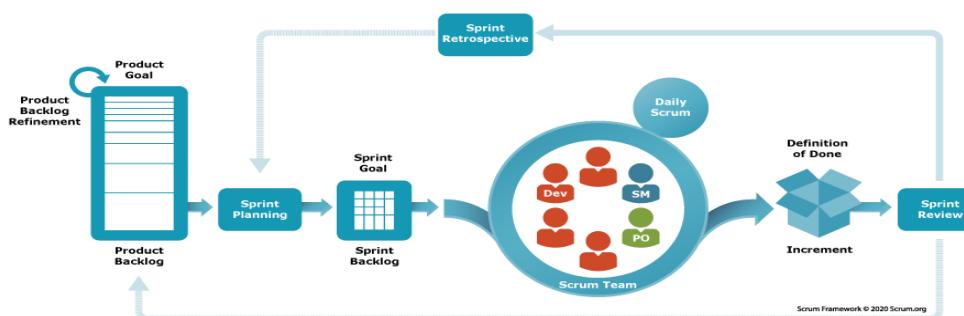
Metode *Agile* merupakan pendekatan dalam pengembangan *software* yang berfokus pada fleksibilitas, kolaborasi, dan respons cepat terhadap perubahan [9]. Pendekatan ini menekankan pentingnya interaksi antar individu daripada bergantung pada alat dan proses, lebih mengutamakan respons terhadap perubahan daripada mengikuti rencana yang telah ditetapkan, serta menjalankan *software* yang berfungsi dengan baik dibandingkan hanya dengan dokumentasi yang terlalu kompleks.

2.2 Scrum

Scrum merupakan kerangka kerja dari metode *Agile* yang digunakan untuk membantu tim, individu, atau organisasi dalam memberikan solusi secara adaptif terhadap permasalahan yang kompleks [10]. Kerangka ini memungkinkan pengembangan sistem dilakukan secara iteratif dan inkremental. Sehingga apabila produk yang dihasilkan pada tahap awal belum memenuhi target kebutuhan, sistem akan dikembangkan lebih lanjut

berdasarkan hasil evaluasi pengguna pada iterasi berikutnya. *Scrum* memiliki tiga elemen utama yaitu peran, artefak, dan acara. Peran terdiri dari *Product Owner*, *Scrum Master*, dan *Development Team*, yang bekerja kolaboratif untuk memastikan nilai produk tercapai. Artefak utama dalam *Scrum* meliputi *product backlog*, *sprint backlog*, dan *increment*, yang dirancang untuk meningkatkan transparansi dan mengukur kemajuan pekerjaan. Sementara itu, acara dalam *Scrum* seperti *Sprint Planning*, *Daily Scrum*, *Sprint Review*, dan *Sprint Retrospective* berfungsi menjaga ritme kerja tim, memfasilitasi inspeksi, dan mendorong perbaikan berkelanjutan.

Sebagaimana ditunjukkan pada Gambar 1, alur kerja dalam metode *Scrum* dimulai dari perencanaan *sprint* berdasarkan daftar prioritas dalam *Product Backlog* yang dibagi menjadi beberapa *sprint backlog*, dilanjutkan dengan pelaksanaan *sprint* oleh tim *Scrum* dalam satuan waktu tertentu, hingga evaluasi hasil melalui *Sprint Review* dan refleksi proses kerja tim pada *Sprint Retrospective*.



Gambar 1. Metode Agile Scrum [11]

2.3 Tinjauan Penelitian Terdahulu

Penerapan metode *Agile Development* dengan model *Scrum* telah banyak digunakan dalam pengembangan sistem informasi, khususnya pada sektor yang membutuhkan fleksibilitas tinggi serta keterlibatan aktif pengguna seperti bidang kesehatan dan pendidikan. *Scrum* memungkinkan proses pengembangan yang iteratif dan terstruktur, sehingga sistem

yang dihasilkan dapat disesuaikan secara bertahap terhadap kebutuhan pengguna.

Dalam bidang kesehatan, Sidiq *et al.* [12] menerapkan *Scrum* dalam pengembangan sistem informasi pelayanan gizi di RSUD Ciamis. Pengembangan dilakukan melalui tiga *sprint* berdurasi 27–30 hari, mencakup *sprint planning*, *daily Scrum*, *sprint review*, dan *sprint retrospective*. Hasilnya menunjukkan bahwa *Scrum* efektif

dalam mengelola *backlog*, pembagian peran tim, serta menghasilkan sistem gizi yang disesuaikan dengan kebutuhan rumah sakit.

Penelitian lain yang dilaporkan pada [13] dan [14], menghasilkan perancang Sistem Informasi Praktik Dokter Gigi berbasis website untuk mengatasi pencatatan manual yang rentan terhadap kesalahan dan keterlambatan pelayanan. Dengan menggunakan metode *Agile* dengan pendekatan *Scrum* untuk pengembangan sistem, penelitian ini juga mengevaluasi performa dan kegunaan sistem menggunakan GTMetrix dan *System Usability Scale* (SUS). Hasilnya menunjukkan bahwa sistem dapat meningkatkan efisiensi dan efektivitas pelayanan pada praktik dokter gigi di daerah Jimbaran.

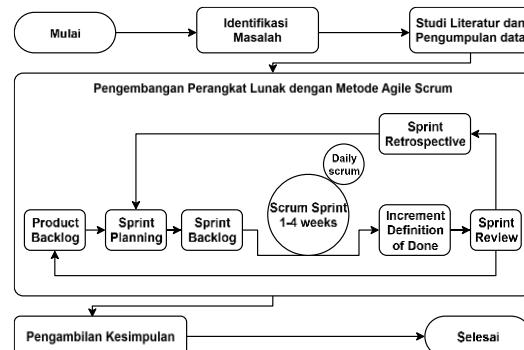
Sementara itu, dalam bidang pendidikan, Widyastuti *et al.* [15] menerapkan *Scrum* dalam pengembangan sistem informasi pengelolaan asrama santri. Dengan menyusun *backlog*, *sprint backlog*, dan pelaksanaan *sprint*, sistem dapat disesuaikan secara bertahap sesuai masukan dari pengguna. Penelitian ini membuktikan bahwa *Scrum* dapat memperkuat kerja sama tim dan mempercepat proses pengembangan sistem dalam lingkungan pendidikan.

Secara umum, penelitian-penelitian tersebut menunjukkan bahwa penerapan *Agile Scrum* merupakan pendekatan yang efektif dalam pengembangan sistem informasi, karena mampu meningkatkan efisiensi kerja tim, memperkuat keterlibatan pengguna, serta menghasilkan sistem yang lebih adaptif terhadap kebutuhan yang terus berkembang.

3. METODOLOGI PENELITIAN

Pada penelitian ini pengembangan sistem informasi administrasi Coass pada Rumah Sakit Pendidikan Gigi dan Mulut menggunakan metode *Agile* dengan kerangka kerja *Scrum*. *Scrum* dipilih karena mampu menghadirkan proses pengembangan yang adaptif, iteratif, dan terstruktur dalam menyikapi perubahan kebutuhan pengguna. Tahapan

pengembangan sistem yang dilakukan mengacu pada urutan proses dalam *Scrum* sebagaimana ditunjukkan pada Gambar 2.



Gambar 2. Proses Pengembangan Sistem

Berikut penjelasan pada Gambar 2 :

Langkah 1. Identifikasi Masalah

Tahap pertama adalah identifikasi masalah, pada tahap ini dilakukan analisis terhadap kendala dalam proses administrasi kegiatan Coass pada RSGM. Beberapa kendala yang ditemukan dari proses yang masih dilakukan secara manual yaitu pencatatan data yang lambat, kurang terstruktur, dan minim transparansi.

Langkah 2. studi literatur dan Pengumpulan data

Tahap kedua mencakup dua aktivitas utama, yaitu studi literatur dan pengumpulan data. Studi literatur dilakukan untuk memperoleh landasan teoritis terkait sistem informasi, metode pengembangan perangkat lunak *Agile Scrum*, serta praktik terbaik dalam pengelolaan administrasi pendidikan klinik. Melalui studi ini, peneliti dapat memahami kerangka konseptual yang relevan dan mengidentifikasi solusi yang telah diterapkan pada penelitian sebelumnya. Selanjutnya, pengumpulan data dilakukan melalui wawancara dan observasi yang dilakukan bersama narasumber serta pihak RSGM. Tujuan dari proses ini adalah untuk memperoleh pemahaman yang lebih mendalam mengenai kebutuhan pengguna serta permasalahan yang dihadapi. Dari hasil pengumpulan data tersebut dibuatlah *user story*, yaitu deskripsi sederhana dari sudut pandang pengguna mengenai fitur yang mereka perlukan.

Langkah 3. Pengembangan Perangkat Lunak Dengan Metode Agile Scrum

Pengembangan perangkat lunak dilakukan dengan metode Agile Scrum dimulai dari penyusunan *product backlog* sebagai daftar kebutuhan sistem. Proses perencanaan *sprint* (*sprint planning*) digunakan untuk memilih *backlog* prioritas ke dalam *sprint backlog* yang akan dikerjakan dalam rentang waktu *sprint* selama 1-4 minggu. Setiap hari selama *sprint* dilakukan *daily scrum* untuk memantau progres dan hambatan. Setelah pengerjaan *backlog* menghasilkan *increment* yang memenuhi kriteria *definition of done*, proses dilanjutkan ke *sprint review* untuk mengevaluasi hasil, dan *sprint retrospective* untuk meninjau proses serta meningkatkan kinerja tim pada *sprint* berikutnya. Seluruh proses berlangsung secara iteratif hingga seluruh *backlog* terselesaikan dan sistem siap digunakan.

Langkah 4. Pengambilan Kesimpulan

Setelah beberapa siklus *sprint* diselesaikan dan sistem telah mencapai kondisi akhir yang diharapkan, dilakukan evaluasi menyeluruh terhadap hasil pengembangan. Proses ini mencakup pengujian akhir sistem, analisis efektivitas fitur, dan penyusunan kesimpulan terhadap implementasi metode *Scrum* dalam penelitian.

3.1 Identifikasi Kebutuhan

Identifikasi kebutuhan sistem dilakukan melalui penyusunan *user story* berdasarkan peran pengguna, seperti admin, mahasiswa Coass, dan dokter. Langkah awal dalam membuat *user story* adalah dengan menentukan fitur-fitur yang akan dibuat berdasarkan hasil observasi dan wawancara pada *users*. Berikut adalah realisasi tahap *user story* pada Tabel 1.

Tabel 1. User Story

Peran	Fitur	Tujuan
Admin	<i>Login/Logout, Dashboard, Update Profile</i>	Mengakses sistem, melihat fitur admin, dan memperbarui data akun

Peran	Fitur	Tujuan
Dokter	Manajemen Dokter, Mahasiswa, Periode, Departemen	Mengelola data entitas sistem secara menyeluruh
	Manajemen <i>Requirement Stase</i> dan <i>Form Penilaian</i>	Mengatur kebutuhan stase dan instrumen penilaian
	Manajemen <i>Logbook</i>	Mengelola dan memverifikasi <i>logbook</i> mahasiswa Coass
	Penilaian Mahasiswa Coass	Memberikan nilai berdasarkan <i>logbook</i> dan <i>form</i> penilaian
Mahasiswa	<i>Login/Logout, Dashboard, Update Profile</i>	Mengakses sistem dan memperbarui data akun
	Verifikasi <i>Logbook</i> Mahasiswa	Memastikan keabsahan <i>logbook</i> kegiatan Coass
	Penilaian Mahasiswa Coass	Memberikan nilai pada mahasiswa sesuai observasi klinis
	Lihat <i>Progress Requirement Stase</i>	Mengetahui capaian kegiatan Coass mahasiswa
	<i>Login/Logout, Dashboard, Update Profile</i>	Mengakses sistem dan memperbarui data akun
	Manajemen <i>Logbook</i>	Mencatat kegiatan harian/mingguan Coass
	Lihat <i>Progress Requirement Stase</i>	Memantau perkembangan kegiatan stase
	Lihat Nilai	Mengetahui hasil penilaian dari dokter dan admin

Tahap berikutnya adalah membuat *product backlog* berdasarkan *user story* yang tertera pada Tabel 1. *Product backlog* ini juga dapat disebut sebagai daftar fitur yang akan dimiliki oleh sistem. Fitur-fitur tersebut diatur dengan cermat dan diberikan tingkatan prioritas untuk pengerjaannya. Berikut pada realisasi dari tahap *product backlog*. Pada Tabel 2.

Tabel 2. Product Backlog

No	Backlog item	Peran Terkait	Prioritas
1	Manajemen Autentikasi (<i>Login/Logout</i>)	Admin, Dokter, Mahasiswa Coass	Tinggi
2	Manajemen Profil Pengguna (<i>Update Profile</i>)	Admin, Dokter, Mahasiswa Coass	Sedang

No	Backlog item	Peran Terkait	Prioritas
3	Pengembangan Dashboard	Admin, Dokter, Mahasiswa Coass	Sedang
4	Manajemen Data Dokter	Admin	Tinggi
5	Manajemen Data Mahasiswa Coass	Admin	Tinggi
6	Manajemen Data Departemen	Admin	Sedang
7	Manajemen Data Periode	Admin	Sedang
8	Manajemen Data Requirement Stase	Admin	Tinggi
9	Manajemen Data Form Penilaian	Admin	Tinggi
10	Manajemen Logbook	Admin Mahasiswa Coass	Tinggi
11	Verifikasi Logbook	Dokter, Admin	Tinggi
12	Penilaian Mahasiswa Coass	Dokter, Admin	Tinggi
13	Melihat Progress Requirement Stase	Dokter, Mahasiswa Coass	Tinggi
14	Melihat Nilai	Mahasiswa Coass	Tinggi

Pada Tabel 2 terdapat 14 *backlog item* dalam *product backlog* Sistem Informasi Administrasi Kegiatan Coass (*Co-Assistant*) pada Rumah Sakit Pendidikan Gigi dan Mulut. Pembuatan *product backlog* pada tahap awal akan membantu tim *Scrum* mendapatkan pemahaman yang jelas tentang pekerjaan yang harus dilakukan.

3.2 Perencanaan Sprint

Tahap ketiga dalam metode *Agile Development* adalah *sprint planning*. Proses ini dilakukan melalui diskusi bersama oleh tim *Scrum*. Selama diskusi, tim *Scrum* harus menetapkan tujuan *sprint* yang akan dijalankan. Selanjutnya, tim *Scrum* akan memilih beberapa *backlog item* yang akan dikerjakan selama *sprint* tersebut. Terakhir, tim *Scrum* akan membahas perencanaan pekerjaan yang perlu dilakukan untuk menyelesaikan *increment* yang memenuhi *definition of done*. Hasil dari *sprint planning* ini berupa *sprint backlog*.

Pada penelitian ini, pengembangan sistem dibagi ke dalam 5 *sprint* selama total 37 hari. Pembagian ini didasarkan pada estimasi awal menggunakan pendekatan *initial velocity estimation*, karena tim belum memiliki data historis produktivitas. Estimasi dilakukan berdasarkan jumlah *backlog* dan kompleksitas teknis dari tiap fitur, lalu diterjemahkan menjadi total 106 *story point*. Pembagian *sprint* diatur agar tiap siklus berisi beban kerja yang proporsional, dan disesuaikan dengan keterbatasan waktu serta komposisi tim pengembang. Ke depan, hasil realisasi *story point* dari masing-masing *sprint* dapat digunakan untuk menyempurnakan estimasi *velocity* pada proyek serupa.

Tabel 3. Sprint Backlog 1

No	Backlog item	Task	Story point
1	<i>Login/Logout</i>	Implementasi sistem <i>login/logout</i> untuk semua <i>users</i>	3
		Menampilkan fitur <i>Dashboard</i> sesuai peran Admin (struktur dasar/ <i>placeholder</i>)	2
		Menampilkan fitur <i>Dashboard</i> sesuai peran Dokter (struktur dasar/ <i>placeholder</i>)	2
		Menampilkan fitur <i>Dashboard</i> sesuai peran Mahasiswa Coass (struktur dasar/ <i>placeholder</i>)	2
3	Manajemen Profil Pengguna	Membuat struktur awal UI untuk halaman <i>Update</i> profil pengguna	2
4	Manajemen Data Departemen	Implementasi CRUD untuk data departemen (tabel <i>departments</i>)	3
5	Manajemen Data Periode	Implementasi CRUD untuk data periode (tabel <i>periods</i>)	3
6	Penilaian Mahasiswa Coass (Dasar)	Desain struktur data dan UI awal untuk input penilaian (tabel <i>form_assessments</i>)	2
		Implementasi CRUD dasar untuk Jenis Skor (tabel <i>score_types</i>)	3

Pada Tabel 3 dapat dilihat *sprint backlog* pertama yang terdiri dari 6 item *backlog*, dibagi menjadi beberapa tugas untuk menyelesaikan fitur. Total *story point* pada *sprint* ini adalah 22.

Tabel 4. Sprint Backlog 2

No	Backlog item	Task	Story point
1	Manajemen Profil Pengguna	Menyelesaikan fitur <i>Update Profil</i> untuk semua peran	2
2	Manajemen <i>Logbook</i> (Mahasiswa)	Implementasi fitur input <i>Logbook</i> oleh Mahasiswa Coass	3
		Menampilkan daftar <i>Logbook</i> yang telah diinput oleh mahasiswa	2
3	Manajemen Data Dokter	Implementasi CRUD untuk data dokter	3
4	Manajemen Data Mahasiswa Coass	Implementasi CRUD untuk data mahasiswa Coass	3
5	Manajemen Data <i>Requirement Stase</i>	Implementasi CRUD dasar untuk data <i>Requirement stase</i>	3
6	Manajemen Data <i>Form Penilaian</i> (Struktur)	Implementasi CRUD untuk pengaturan <i>form penilaian</i> (<i>score_form_settings</i>)	3
		Implementasi CRUD untuk header <i>form penilaian</i> (<i>score_form_setting_headers</i>)	3
7	Penilaian Mahasiswa Coass (Input Detail)	Mengembangkan <i>form</i> input penilaian	3

Pada Tabel 4 dapat dilihat *sprint backlog* kedua dengan total 25 *story point*. Fokus utamanya pada penyelesaian Manajemen Profil Pengguna, *Logbook* Mahasiswa, Manajemen Data Dokter, Coass, *Requirement Stase*, serta pengembangan awal *Form Penilaian*.

Tabel 5. *Sprint Backlog 3*

No	Backlog item	Task	Story point
1	Manajemen <i>Logbook</i>	Menghubungkan input <i>Logbook</i> dengan <i>Requirement_id</i>	2
		<i>Update Requirement_progres</i> s berdasarkan <i>Logbook</i> (dasar)	2
		Menampilkan detail <i>Logbook</i> terkait <i>progress</i> (dasar)	2
2	Manajemen Data <i>Requirement Stase</i>	Validasi relasi <i>Requirements</i> dan <i>departments</i>	2

No	Backlog item	Task	Story point
3	Manajemen Data <i>Form Penilaian</i> (Detail)	Implementasi CRUD <i>score_form_setting_sub_headers</i> (dasar)	3
4	Melihat <i>Progress Requirement</i> (Mahasiswa)	Halaman <i>progress Requirement</i> mahasiswa	2
5	Penilaian Mahasiswa Coass (Proses dan Log Aktivitas)	Implementasi penyimpanan detail skor ke (<i>form_assessment_details</i>)	3
		Perhitungan <i>total_score</i> pada <i>form_assessments</i> (dasar)	2
		Implementasi dasar <i>assessment_logs</i>	3

Pada Tabel 5 dapat dilihat *sprint backlog* ketiga dengan total 21 *story point*. Fokus *sprint* ini adalah integrasi *Logbook* dengan *Requirement*, pendetailan *Form Penilaian*, pengembangan fitur *progress mahasiswa*, dan implementasi awal penyimpanan skor penilaian.

Tabel 6. *Sprint Backlog 4*

No	Backlog item	Task	Story point
1	Verifikasi <i>Logbook</i>	Implementasi fitur verifikasi <i>Logbook</i> (Dokter/Admin)	2
		Indikator visual untuk <i>Logbook</i> yang perlu diverifikasi	1
2	Dashboard (Mahasiswa)	Menampilkan ringkasan data di <i>Dashboard</i> Mahasiswa	2
3	Melihat <i>Progress Requirement</i>	Halaman <i>progress Requirement</i> untuk Dokter	2
		Halaman <i>progress Requirement</i> untuk Admin	2
4	Manajemen Data Dokter	Implementasi <i>import excel</i> untuk tambah data dokter	2
5	Manajemen Data Mahasiswa Coass	Implementasi <i>import excel</i> untuk tambah data mahasiswa Coass	2
6	Penilaian Mahasiswa Coass	Implementasi logika perhitungan <i>final_scores</i>	3
		Penyimpanan hasil kalkulasi nilai dan	2

No	Backlog item	Task	Story point
		integrasi dengan Requirement_progress	
7	Melihat Nilai (Mahasiswa)	Menampilkan nilai akhir dan rincian dasar untuk Mahasiswa	2

Pada Tabel 6 dapat dilihat *sprint backlog* keempat dengan total 20 *story point*. Fokusnya mencakup Verifikasi *Logbook* oleh dosen dan admin, *Dashboard Mahasiswa*, laporan *progress* untuk Dokter dan Admin, impor data menggunakan Excel, kalkulasi otomatis Nilai Akhir, serta fitur bagi mahasiswa untuk melihat hasil evaluasi.

Tabel 7. *Sprint Backlog 5*

No	Backlog item	Task	Story point
1	<i>Dashboard</i> (Dokter dan Admin)	Menampilkan ringkasan data di <i>Dashboard Dokter</i>	2
		Menampilkan ringkasan data di <i>Dashboard Admin</i>	2
2	Pengujian Sistem	Pengujian modul Autentikasi, Profil, <i>Logbook</i>	3
		Pengujian modul Master Data, Pelaporan <i>Progress</i>	3
		Pengujian modul Penilaian dan Kalkulasi Nilai	3
3	Perbaikan sistem	Perbaikan <i>Bug</i> dan performa Website	5

Pada Tabel 7 dapat dilihat *sprint backlog* kelima dengan total 18 *story point*. Sprint ini mencakup *Dashboard Dokter* dan Admin, Pengujian Sistem, serta Perbaikan Sistem. Fokus utamanya adalah penyajian ringkasan data, pengujian modul utama, dan peningkatan stabilitas serta performa aplikasi.

3.3 Pelaksanaan Sprint

Daily scrum merupakan tahapan di mana tim menginformasikan pembaruan mengenai apa yang akan dikerjakan, apa yang sedang kerjakan dan apa yang telah diselesaikan. Dalam setiap *daily scrum*, anggota tim menjelaskan progres sejak pertemuan terakhir, rencana kerja untuk periode selanjutnya, serta hambatan atau

masalah yang mereka hadapi. *Daily scrum* bertujuan untuk memastikan semua anggota tim terkoordinasi dengan baik dan dapat berkolaborasi secara efektif untuk menyelesaikan *sprint backlog*.

Dalam pengembangan Sistem Informasi Administrasi Kegiatan Coass berbasis web dengan metode *Agile Development* model *Scrum* ini, tim secara konsisten melaksanakan sesi *daily scrum* pada setiap *sprint*. Total lima *sprint* dilakukan selama siklus pengembangan, dan masing-masing *sprint* mencakup 6 hingga 9 sesi *daily scrum*, tergantung pada kompleksitas dan durasi *sprint* yang dilaksanakan. Ringkasan pelaksanaan *daily scrum* pada masing-masing *sprint* dapat dilihat pada Tabel 8.

Tabel 8. Ringkasan *Daily Scrum*

Sprint	Waktu	Task	Story Point	Remaining Story Point
1	17 - 23 Februari 2025	Autentikasi, <i>Dashboard</i> awal, Master Data Dasar	22	84
2	24 Februari - 3 Maret 2025	Profil Pengguna, Input & Daftar <i>Logbook</i> , Form Penilaian Dinamis	25	59
3	5 - 10 Maret 2025	Integrasi <i>Logbook</i> dengan <i>Requirement</i> , <i>Progress</i> Mahasiswa, Penilaian Mahasiswa	21	38
4	13 - 21 Maret 2025	Verifikasi <i>Logbook</i> , <i>Dashboard</i> , Kalkulasi Nilai Akhir	20	18
5	24 - 26 Maret dan 9 - 12 April 2025	Pengujian Sistem, Perbaikan <i>Bug</i> , Finalisasi Tampilan dan <i>Dashboard</i>	18	0

Selama pelaksanaan *daily scrum*, tim mendokumentasikan status pekerjaan menggunakan pendekatan kolaboratif, mencakup kolom *To Do*, *In Progress*, *Done*,

serta kendala (*Problems*). Format ini memudahkan tim untuk melakukan evaluasi harian dan penyesuaian strategi implementasi fitur secara iteratif.

Pelaksanaan *daily scrum* secara konsisten memungkinkan proses pengembangan sistem berlangsung secara iteratif, terukur, dan adaptif terhadap perubahan kebutuhan. Pendekatan ini terbukti efektif dalam mendukung kolaborasi tim serta mempercepat penyelesaian *backlog* secara efisien.

3.4 Evaluasi

Sprint Review merupakan sesi evaluasi yang dilakukan di akhir setiap Sprint dalam metode *Scrum*. Pada sesi ini, tim akan menunjukkan hasil pekerjaan mereka guna memperoleh masukan serta menentukan langkah berikutnya. Hasil untuk *sprint review* dapat dilihat pada Tabel 9.

Tabel 9. *Sprint Review*

Sprint	Hasil Sprint	Masukan
1	Menyelesaikan 6 <i>backlog item</i> (22 SP). Fokus pada fondasi sistem: <i>login/logout</i> , <i>dashboard</i> awal, manajemen data dasar, dan desain awal sistem penilaian.	Tambahkan fitur <i>remember me</i> saat <i>login</i> .
2	Menyelesaikan 7 <i>backlog item</i> (25 SP). Fitur utama: <i>update profil</i> , <i>input</i> dan <i>daftar Logbook</i> , <i>CRUD Dokter</i> , <i>Mahasiswa</i> , <i>Requirement Stase</i> , serta struktur awal <i>form penilaian</i> .	Perjelas alur antara pengaturan nilai, <i>header</i> , dan <i>sub-header form</i> penilaian.
3	Menyelesaikan 5 <i>backlog item</i> (21 SP). Fokus: integrasi <i>Logbook</i> dengan <i>Requirement</i> , <i>progress mahasiswa</i> , <i>sub-header form</i> penilaian, serta penyimpanan skor dan log penilaian.	Tampilan <i>form</i> perlu menyesuaikan dengan format asli.
4	Menyelesaikan 7 <i>backlog item</i> (20 SP). Fitur: verifikasi <i>Logbook</i> , <i>dashboard mahasiswa</i> , <i>progress mahasiswa</i> , <i>import data Excel</i> , penilaian akhir, dan tampilan nilai.	Tambah detail <i>progress</i> per departemen dan log penilaian.
5	Menyelesaikan 3 <i>backlog item</i> (18 SP). Fokus: finalisasi <i>dashboard</i> , pengujian sistem, dan perbaikan <i>bug</i> .	Tingkatkan konsistensi dan <i>responsive</i> tampilan.

Tahapan terakhir dalam metode *Agile Scrum* adalah *Sprint Retrospective* adalah tahap di mana tim melakukan evaluasi terhadap cara pelaksanaan *sprint* untuk mengidentifikasi hal-hal yang berjalan baik maupun yang perlu diperbaiki. Hasil untuk *sprint retrospective* dapat dilihat pada Tabel 10.

Tabel 10. *Sprint Retrospective*

Sprint	Sprint Retrospective
1	<i>Sprint</i> pertama terlaksana dengan baik. Tim berhasil menyelesaikan seluruh <i>backlog</i> tepat waktu. Komunikasi antar tim berjalan efektif, namun integrasi komponen aplikasi dan validasi antarmuka pengguna masih perlu ditingkatkan. <i>Sprint</i> selanjutnya difokuskan pada pengujian fungsional dan validasi alur sistem secara menyeluruh.
2	Tim menyelesaikan seluruh fitur utama sesuai rencana. Meskipun proses pengembangan berjalan lancar, kurangnya perencanaan alur data antar modul serta dokumentasi teknis berdampak pada menurunnya efisiensi kerja tim. Untuk perbaikan, tim menyepakati kejelasan logika sistem dan peningkatan dokumentasi sebelum pengembangan dimulai.
3	Tim berhasil mengintegrasikan <i>logbook</i> dengan <i>progress requirement</i> . Tantangan muncul pada penyusunan logika pembaruan dan optimasi <i>query</i> . Tim menindaklanjuti dengan pembagian <i>debugging</i> yang lebih sistematis serta percepatan validasi teknis sejak awal <i>sprint</i> .
4	Tim menyelesaikan fitur verifikasi <i>logbook</i> , <i>progress requirement</i> , hingga perhitungan nilai akhir mahasiswa. Kompleksitas logika dan laporan menjadi kendala utama. Tim merespons dengan memperkuat validasi logika bisnis sejak awal serta membagi tanggung jawab pengujian secara merata.
5	<i>Sprint</i> terakhir difokuskan pada penyempurnaan tampilan dan pengujian sistem. Meskipun berlangsung saat libur nasional, tim tetap menjaga koordinasi harian. Koordinasi lintas modul sempat menjadi tantangan, namun sistem berhasil diselesaikan dalam kondisi stabil dan konsisten.

Evaluasi sistem dilakukan secara berkala melalui *Sprint Review* dan *Sprint Retrospective* di setiap akhir *sprint*. *Sprint Review* menilai ketercapaian *backlog* dan fungsionalitas sistem, sedangkan *Sprint Retrospective* merefleksikan kekuatan dan kendala selama proses pengembangan. Dari *Sprint 1* hingga *Sprint 5*, seluruh

backlog berhasil diselesaikan sesuai rencana, menandakan perencanaan yang realistik dan kerja tim yang efektif. Beberapa hambatan seperti integrasi antar modul, alur data, dan kompleksitas *query* disikapi dengan pembagian tugas *debugging* yang lebih sistematis serta validasi logika sejak awal. Evaluasi ini mendorong peningkatan kualitas sistem secara berkelanjutan dan memperkuat kolaborasi dalam tim.

3.5 Struktur Tim Scrum dan Time-Boxing Aktivitas

Dalam penelitian ini, proses pengembangan sistem informasi administrasi *Coass* pada RSGM dilakukan oleh tim yang menerapkan struktur peran sesuai kerangka kerja *Scrum*. Pembagian peran dalam tim sebagai berikut:

1. *Product Owner* (PO): Bertanggung jawab menetapkan prioritas *backlog* dan memastikan fitur yang dikembangkan sesuai dengan kebutuhan pengguna. Peran ini diwakili oleh narasumber dari pihak RSGM..
2. *Scrum Master* (SM): Bertugas memastikan proses Scrum berjalan sesuai prinsip dan memfasilitasi tim dalam mengatasi hambatan. Peran ini dijalankan oleh dosen pembimbing.
3. *Development Team*: Terdiri dari tiga orang pengembang (Dev1, Dev2, Dev3) yang berkolaborasi dalam menyelesaikan *backlog* secara bertahap di setiap *sprint*.

Pengerjaan dilakukan secara inkremental, artinya setiap fitur dibangun dan disempurnakan bertahap dari *sprint* pertama hingga *sprint* kelima. Untuk menjaga ritme kerja dan efektivitas kolaborasi, tim mengikuti pedoman durasi aktivitas (*time-boxing*) berdasarkan panduan *Scrum Guide*. Rincian *time-boxing* tersebut dapat dilihat pada Tabel 11.

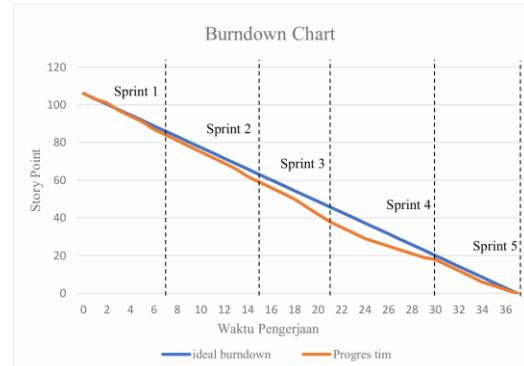
Tabel 11. *Time-Boxing* Aktivitas Scrum

Aktivitas Scrum	Durasi Maksimal	Frekuensi Pelaksanaan
Sprint	± 7-8 hari kerja per sprint	Sekali per siklus (total 5 kali)

Aktivitas Scrum	Durasi Maksimal	Frekuensi Pelaksanaan
<i>Sprint Planning</i>	Maksimal 4 jam	Awal setiap <i>sprint</i>
<i>Daily Scrum</i>	Maksimal 15 menit/hari	Setiap hari kerja dalam <i>sprint</i>
<i>Sprint Review</i>	Maksimal 2 jam	Akhir setiap <i>sprint</i>
<i>Sprint Retrospective</i>	Maksimal 1,5 jam	Setelah <i>Sprint Review</i>

4. HASIL DAN PEMBAHASAN

Keberhasilan metode *Agile Scrum* dalam pengembangan sistem informasi administrasi kegiatan *Coass* dapat dilihat pada *Burndown Chart* pada Gambar 3.



Gambar 3. *Burndown Chart*

Garis biru menunjukkan penurunan *story point* ideal bila pekerjaan berjalan konsisten, sementara garis oranye merepresentasikan progres aktual harian tim. Berdasarkan grafik tersebut, dapat dianalisis durasi pengerjaan setiap *sprint* sebagai berikut:

Tabel 11. Analisis Durasi Sprint

Sprint	Waktu	Story Point	Analisis
Sprint 1	17 - 23 Februari 2025	22	Progres sedikit tertinggal di awal <i>sprint</i> karena proses adaptasi terhadap peran <i>Scrum</i> dan tools. Namun, seluruh <i>backlog</i> terselesaikan tepat waktu
Sprint 2	24 Februari -	25	Penurunan <i>backlog</i> berlangsung lebih lambat di tengah <i>sprint</i> karena item

Sprint	Waktu	Story Point	Analisis
	3 Maret 2025		<i>backlog</i> mencakup validasi relasi data dan struktur <i>form</i> penilaian yang cukup kompleks
Sprint 3	5 - 10 Maret 2025	21	Penyelesaian <i>backlog</i> meningkat signifikan, progres aktual melampaui garis ideal. Hal ini mencerminkan peningkatan ritme kerja serta pemahaman tim terhadap struktur sistem dan mekanisme <i>sprint</i>
Sprint 4	13 - 21 Maret 2025	20	Progres sempat stagnan di pertengahan <i>sprint</i> karena <i>backlog</i> melibatkan logika perhitungan nilai akhir dan validasi <i>logbook</i> yang kompleks yang membutuhkan sinkronisasi antar modul.
Sprint 5	24 - 26 Maret dan 9 - 12 April 2025	18	Progres tetap stabil meskipun dilaksanakan bertepatan dengan masa libur nasional. Fokus utama pada pengujian sistem, perbaikan <i>bug</i> , dan penyempurnaan antarmuka pengguna.

Berdasarkan data pada Tabel 11, kecepatan penyelesaian *backlog* sangat dipengaruhi oleh adaptasi awal, kompleksitas teknis, serta koordinasi tim. Performa terbaik terlihat pada Sprint 3, menunjukkan efektivitas pendekatan *Scrum* dalam membentuk ritme kerja yang adaptif dan berorientasi pada penyelesaian *backlog* secara terukur.

Penyelesaian *backlog* secara keseluruhan menunjukkan bahwa sistem informasi yang dibangun telah berfungsi sesuai kebutuhan. Sistem ini mampu menggantikan proses administrasi manual dalam kegiatan *coass* yang sebelumnya rentan terhadap inefisiensi, kehilangan data, dan kurangnya transparansi. Dengan sistem berbasis *website*, proses pencatatan dan pelaporan menjadi lebih efisien,

terorganisasi, dan dapat diakses dengan mudah oleh pihak terkait.

Kendala utama selama pengembangan adalah integrasi antar modul dan alur data yang belum matang, dapat dilihat dalam *Sprint Retrospective* pada Tabel 10. *Scrum* yang iteratif dan adaptif memungkinkan tim mengatasinya melalui *daily scrum* dan diskusi lintas peran. Evaluasi rutin menghasilkan perbaikan seperti *debugging* antar modul, penyesuaian alur data, dan validasi ulang logika bisnis. Pembelajaran penting dari proses ini mencakup pentingnya perencanaan alur data sejak awal, komunikasi terbuka dalam tim, dan fleksibilitas *Scrum* dalam menyesuaikan dengan kompleksitas sistem seperti di RSGM.

Hasil pada penelitian ini sejalan dengan temuan pada beberapa penelitian terdahulu. Penelitian yang dilakukan oleh Sidiq *et al.* [12] membuktikan bahwa metode *Scrum* efektif dalam mengelola *backlog* dan pembagian peran tim dalam pengembangan sistem informasi pelayanan gizi. Hal tersebut juga tercermin dalam proyek ini, yang berhasil menyelesaikan 106 *story point* secara tepat waktu melalui tahapan *Scrum* yang terstruktur. Tantangan integrasi antar modul yang muncul selama pengembangan sistem ini memiliki kemiripan dengan hambatan teknis yang dijelaskan oleh Widayastuti *et al.* [15], yang menyatakan pentingnya pelaksanaan *sprint retrospective* dalam menyelesaikan kendala teknis pada sistem informasi asrama. Di sisi lain, hasil penelitian pada [13] dan [14] menunjukkan bahwa penerapan *Scrum* dalam pengembangan sistem informasi praktik dokter gigi mampu meningkatkan efisiensi dan efektivitas pelayanan. Temuan tersebut sejalan dengan hasil pengembangan sistem informasi ini, yang menggantikan proses manual dalam administrasi kegiatan *Coass* dan meningkatkan efisiensi pencatatan serta transparansi informasi. Meskipun konteks ketiga penelitian tersebut berbeda, seluruhnya menunjukkan bahwa metode *Scrum* efektif dalam menghasilkan sistem informasi yang adaptif dan sesuai

kebutuhan pengguna. Perbedaan utama pada proyek ini terletak pada tingkat kompleksitas sistem yang lebih tinggi, karena mencakup berbagai aktivitas administrasi, melibatkan banyak peran, serta alur data yang saling terintegrasi. Hal ini menunjukkan bahwa metode *Scrum* mampu menangani pengembangan sistem yang kompleks secara efektif, dibuktikan melalui penyelesaian 106 *story point* secara tepat waktu dan kemampuan tim dalam mengatasi hambatan teknis melalui praktik *daily scrum*, *sprint review* dan *sprint retrospective* yang berkelanjutan.

5. KESIMPULAN

Metode *Agile Scrum* terbukti menjadi pendekatan yang efektif dalam pengembangan sistem informasi administrasi *Coass* di lingkungan rumah sakit pendidikan yang dinamis. Keberhasilan proyek ini ditandai dengan penyelesaian 106 *story point* dalam lima *sprint*. Hal ini menunjukkan bahwa kerangka kerja iteratif *Scrum* mampu membantu tim dalam menghadapi tantangan teknis, seperti integrasi antar modul dan kompleksitas logika bisnis.

Selain meningkatkan efisiensi proyek, fleksibilitas *Scrum* juga memungkinkan pengembangan sistem yang tidak hanya memenuhi kebutuhan teknis, tetapi juga selaras dengan alur kerja pengguna akhir (*end user*). Dengan demikian, penelitian ini memberikan kontribusi praktis berupa model pengembangan adaptif yang dapat direplikasi oleh institusi pendidikan klinis lain dalam mempercepat transformasi digital pada proses administrasi akademis dan klinis.

Melihat potensi keberlanjutan proyek perangkat lunak ini, sangat memungkinkan *backlog* baru akan muncul seiring perkembangan kebutuhan pengguna atau integrasi sistem lanjutan. Oleh karena itu, histori penggeraan sebelumnya seperti durasi penggeraan per *sprint*, jumlah *story point* harian yang realistik, dan kompleksitas *backlog* dapat dijadikan referensi untuk menyusun estimasi *sprint* berikutnya. Data dari lima *sprint* sebelumnya sebaiknya

dijadikan dasar pengambilan keputusan dalam menetapkan kapasitas *sprint*, memetakan risiko teknis, serta mengantisipasi hambatan dalam iterasi pengembangan lanjutan.

6. DAFTAR PUSTAKA

- [1] Rizki, A. F., Fadilla, A. N., Rahman, Y. 2020. "Perancangan Aplikasi Mahasiswa Coass Kedokteran Gigi Di Kota Bandung." *EProceedings of Art & Design*, Vol. 7, No. 3.
- [2] Konsil Kedokteran Indonesia. 2015. Standar Kompetensi Dokter Gigi Indonesia. Jakarta: KKI, Hal. 26–27.
- [3] Kementerian Pendidikan dan Kebudayaan Republik Indonesia. 2021. Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 8 Tahun 2021 Tentang Rumah Sakit Perguruan Tinggi Negeri.
- [4] Mathias, A. A. P. 2023. "Pembangunan Sistem Informasi Logbook Koas Fakultas Kedokteran Universitas XYZ". *Skripsi*. Universitas Atma Jaya Yogyakarta.
- [5] Nuryanis, S., Anjani, D., Yuliani, R. 2022. "Perancangan Sistem Informasi Penilaian COASS Dokter pada RSUD Pasar Rebo." *Jurnal Riset dan Aplikasi Mahasiswa Informatika (JRAMI)*, Vol. 3, No. 04, Hal. 676–682.
- [6] Pertiwi, T. A., Luchia, N. T., Sinta, P., Dahlia, A., Fachrezi, I. R., Aprinastya, R., Hamzah, M. L. 2023. "Perancangan dan Implementasi Sistem Informasi Absensi Berbasis Web Menggunakan Metode Agile Software Development." *Jurnal Testing dan Implementasi Sistem Informasi*, Vol. 1, No. 1, Hal. 53–66.
- [7] Heriyawan, I., Hayati, U., Nurdyawan, O. 2023. "Rancang Bangun Sistem Informasi Akuntansi Menggunakan CodeIgniter dengan Metode Scrum (Studi Kasus: PT Surya Marga Sarana)." *Jurnal Mahasiswa Teknik Informatika*, Vol. 7, No. 2.
- [8] Ariesta, A., Dewi, Y. N., Sariyah, F. A., Fibriany, F. W. 2021. "Penerapan Metode Agile dalam Pengembangan Application Programming Interface System pada PT XYZ." *Seminar Sistem Informasi STMIK Nusa Mandiri*, Jakarta.
- [9] Cunningham, W., van Bennekum, A., Cockburn, A., Beedle, M., Fowler, M.,

- Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Schwaber, K., Sutherland, J., Thomas, D. 2001. *Principles Behind the Agile Manifesto*. Tautan: <https://agilemanifesto.org>
- [10] Schwaber, K., Sutherland, J. 2020. The Scrum Guide. Tautan: <https://Scrumguides.org/>
- [11] Scrum.org. 2023. Scrum Framework Diagram (Updated 29 September 2023). Diakses dari: <https://Scrumorg-website-prod.s3.amazonaws.com/drupal/inline-images/2023-09/Scrum-framework-9.29.23.png>
- [12] Sidiq, M., Supriatman, R. D., Firdaus, E. A., & Suburdjati, B. A. 2024. Perancangan Arsitektur Sistem Informasi Menggunakan Metode Agile Dengan Kerangka Kerja Scrum Pada Pelayanan Instalasi Gizi RSUD Ciamis. *Nuansa Informatika*, Vol. 18, No. 1, hal. 53-67. <https://doi.org/10.25134/ilkom.v18i1.52>
- [13] Christshanty, D., Putra Pande, P. A., Indra ER, N., Purnama, F. 2024, "Development of a Dental Practice Medical Record Using The Scrum Methodology," *Journal of Electrical, Electronics and Informatics (JEEI)*, vol. 8, no. 1, Hal. 1–7.
- [14] Polvi, I K. Y. P., Sucahya W., I P. G. R., Indra ER, N., Purnama, F. 2024., "Analysis of Dentist Practice Information System with System Usability Scale (SUS) Method," *Journal of Electrical, Electronics and Informatics (JEEI)*, vol. 8, no. 2, Hal. 40–46.
- [15] Widyastuti, R., Hartati, T., & Supriyadi, B. 2021. Penerapan Scrum pada Perancangan Sistem Informasi Rumah Santri Indonesia. *Justek: Jurnal Sains dan Teknologi*, Vol. 13, No. 1, hal. 39–47.