

OAJIS

Open Access
Journal of
Information
Systems

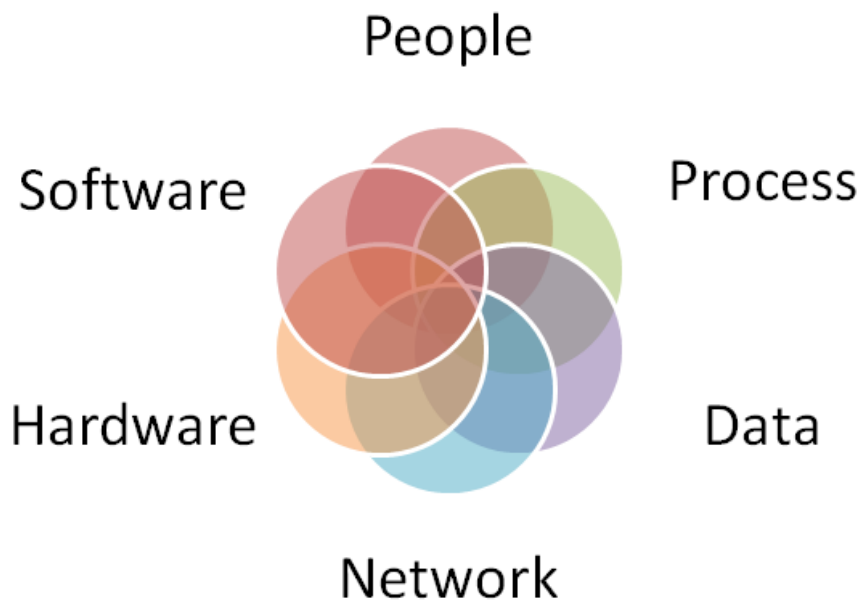
is.its.ac.id/pubs/oajis/

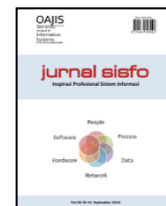
ISSN 1979-3979



jurnal sisfo

Inspirasi Profesional Sistem Informasi





Pimpinan Redaksi

Sholiq (Institut Teknologi Sepuluh Nopember)

Dewan Redaksi

Reny Nadlifatin (Institut Teknologi Sepuluh Nopember)

Tining Haryanti (Universitas Muhammadiyah Surabaya)

Faizal Mahananto (Institut Teknologi Sepuluh Nopember)

Rizal Risnanda Utama (Institut Teknologi Sepuluh Nopember)

Dimas Agung Perkasa (Institut Teknologi Sepuluh Nopember)

Monica Widiarsi (Universitas Surabaya)

Anjik Sukmaaji (Universitas Dinamika)

Devi Septiani (Universitas Brawijaya)

Tata Pelaksana Usaha

Rachmatina Retno Septiani

Sekretariat

Departemen Sistem Informasi – Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember (ITS) – Surabaya

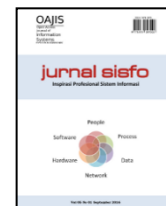
Telp. 031-5999944 Fax. 031-5964965

Email: editor@jurnalsisfo.org

Website: <http://jurnalsisfo.org>

Jurnal SISFO juga dipublikasikan di *Open Access Journal of Information Systems* (OAJIS)

Website: <http://is.its.ac.id/pubs/oajis/index.php>



Mitra Bestari

Prof. Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D
(Institut Teknologi Sepuluh Nopember)

Dr. Jusak (James Cook University, Singapore)

Dr. Muhammad Ainul Yaqin, S.Si., M.Kom. (UIN Maulana Malik Ibrahim)

Dr. Bambang Setiawan, S.Kom., M.T. (Institut Teknologi Sepuluh Nopember)

Dr. Feby Artwodini, S.Kom., M.T. (Institut Teknologi Sepuluh Nopember)

Arif Wibisono, S.Kom., M.Sc., Ph.D. (Institut Teknologi Sepuluh Nopember)

Dr. Agus Subhan Akbar, S.Kom, M.Kom. (Universitas Islam Nahdlatul Ulama Jepara)

Ir. Nunik Endah Sulistiyawati, M.T. (TNI AL, Indonesia)

Dhiani Tresna Absari, S.T. M.Kom. (Universitas Surabaya)

Muhamad Amirul Haq, S.T., M.Sc. (Universitas Muhammadiyah Surabaya)

Ronny Trian Surbakti S.IP., M.M. (Universitas Katolik Parahyangan)

Agus Dwi Purwolastono, SE, M.Acc, Ak. (Institut Teknologi Sepuluh Nopember)



Daftar Isi

Analisis Sentimen dan Korelasi Berita Saham pada Platform Trading Gotrade terhadap Perubahan Harga Saham

Christoforus Fachrasya, Retno Aulia Vinarti, Faizal Mahananto, Amalia Utamima, Renny Pradina Kusumawardani 1

Proyek Pengembangan Notares Website Manajemen Keuangan Cerdas dengan Metode Scrum Agile

Rahmat Ramadhan Permana, Agung Budi Prasetya, Farhan Adika Suwardana, Arayzi Rayyansyah, Arjuna Putra Kharisma, Sholiq Sholiq, Rizal Risnanda Hutama 12

A Bibliometric Analysis of Digital Business Models: Comparative Insights Before and After COVID-19

Febby Candra Pratama 32

Mapping a Decade of Digital Twin Research: Trends, Thematic Evolution, and China's Strategic Lead

Radityo Prasetyanto Wibowo, Prasasti Karunia Farista Ananto, Eugenia Indrawan 46

Pengaruh Fitur Social Media Marketing terhadap Keputusan Pembelian Brand Fashion

Mudjahidin Mudjahidin, Ahmad Ikhsan, Talitha Firyal Ghina Nuha 61

Penyusunan Disaster Recovery Planning (DRP) Pada Data Center Unusa Menggunakan NIST 800-34

Riko Adi Kurniawan, Endang Sulistiyani 81

Application of Gamification to Enterprise Systems: A Systematic Literature Review

Tri Puspa Rinjeni, Mahendrawathi ER 97



Halaman ini sengaja dikosongkan

Proyek Pengembangan Notares *Website* Manajemen Keuangan Cerdas dengan Metode *Scrum Agile*

Rahmat Ramadhan Permana, Agung Budi Prasetya*, Farhan Adika Suwardana, Arayzi Rayyansyah,
Arjuna Putra Kharisma, Sholiq Sholiq, Rizal Risnanda Utama

Departemen Sistem Informasi, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember

Abstract

This study aims to develop Notares, a smart financial management website designed to help students manage expenses and income effectively. In its development, the Agile method with the Scrum framework was applied to enable rapid iteration, optimal team collaboration, and adaptability to dynamic user needs. The development process includes the stages of collecting user stories, compiling a product backlog, planning and implementing sprints, to evaluating and testing features. As a result, the Notares website was successfully developed with core features that consumed a total of 79 hours of work time in three sprint periods (of the five planned sprints). The Notares website provides main features such as financial recording, report visualization, investment recommendations, and smart notifications. The evaluation shows that the Scrum approach can support the achievement of project targets efficiently and provide flexibility in feature development. The novelty of this study lies in the application of the Agile method with the Scrum framework in the context of developing a financial management website for students, which has not yet been used as the main approach in similar studies.

Keywords: *Financial Management, Agile, Scrum, Website Development, Project Management*

Abstrak

Studi ini bertujuan mengembangkan Notares, sebuah website manajemen keuangan cerdas yang dirancang untuk membantu mahasiswa dalam mengelola pengeluaran dan pendapatan secara efektif. Dalam pengembangannya, metode Agile dengan kerangka kerja *Scrum* diterapkan untuk memungkinkan iterasi yang cepat, kolaborasi tim yang optimal, serta kemampuan adaptasi terhadap kebutuhan pengguna yang dinamis. Proses pengembangan mencakup tahapan pengumpulan *user stories*, penyusunan *product backlog*, perencanaan dan pelaksanaan *sprint*, hingga evaluasi dan pengujian fitur. Hasilnya, *website* Notares berhasil dikembangkan dengan fitur inti yang mengonsumsi total waktu pengerjaan 79 jam dalam tiga periode *sprint* (dari lima *sprint* yang direncanakan). *Website* Notares menyediakan fitur utama seperti pencatatan keuangan, visualisasi laporan, rekomendasi investasi, dan notifikasi cerdas. Evaluasi menunjukkan bahwa pendekatan *Scrum* mampu mendukung pencapaian target proyek secara efisien serta memberikan fleksibilitas dalam pengembangan fitur. Keterbaruan dari penelitian ini terletak pada penerapan metode *Agile* dengan kerangka kerja *Scrum* dalam konteks pengembangan *website* manajemen keuangan untuk mahasiswa, yang masih belum dijadikan pendekatan utama dalam studi sejenis.

*Corresponding Author

Email address: agungbudiprasetya9@gmail.com (Agung Budi Prasetya)
<https://doi.org/10.24089/j.sisfo.2025.07.002>

Kata kunci: Manajemen Keuangan, Agile, Scrum, Pengembangan Website, Manajemen Proyek

© 2025 Jurnal SISFO.

Histori Artikel: Disubmit 03-07-2024; Direvisi 27-04-2025; Diterima 02-05-2025; Tersedia online 31-05-2025

1. Pendahuluan

Masalah perilaku keuangan mahasiswa di Indonesia semakin menonjol seiring dengan perkembangan zaman. Teknologi yang berkembang pesat kini memegang peranan penting dalam kehidupan sehari-hari. Gaya hidup masyarakat juga mengalami peningkatan, yang berdampak pada perilaku konsumsi yang tidak stabil, terutama di kalangan generasi muda. Salah satu solusi untuk mengatasi sifat konsumtif adalah dengan menabung, yang merupakan metode efektif untuk mengurangi perilaku boros. Menabung tidak perlu dalam jumlah besar, cukup mengalokasikan beberapa persen dari pendapatan yang diterima [1].

Penelitian yang dilakukan oleh Natalia et al. [2] pada tahun 2019 menunjukkan bahwa pengelolaan uang mahasiswa dapat dikelompokkan berdasarkan tiga kategori pendapatan bulanan: kurang dari Rp1.000.000,00 antara Rp1.000.000,00-Rp1.500.000,00 dan lebih dari Rp1.500.000,00. Mahasiswa dengan pendapatan kurang dari Rp1.000.000,00 memiliki pengelolaan keuangan yang rendah dengan rata-rata 43,1% karena terbatasnya uang yang mereka terima, sehingga tidak cukup untuk memenuhi kebutuhan mereka. Mahasiswa dengan pendapatan antara Rp1.000.000,00-Rp1.500.000,00 juga berada pada kategori pengelolaan keuangan rendah, dengan rata-rata 46%. Meskipun sedikit lebih baik daripada mahasiswa dengan pendapatan kurang dari Rp1.000.000,00 mereka masih menghadapi tantangan dalam perencanaan, penyimpanan, penggunaan, dan pencatatan keuangan. Mahasiswa yang memiliki pendapatan lebih dari Rp1.500.000,00 juga menunjukkan pengelolaan keuangan yang rendah dengan rata-rata 44%, karena kecenderungan untuk lebih boros akibat pendapatan yang lebih besar yang menyebabkan peningkatan pengeluaran. Secara keseluruhan, pengelolaan keuangan masih menjadi tantangan bagi mahasiswa di semua kategori pendapatan [2].

Menyadari permasalahan umum yang dihadapi oleh mahasiswa ini, penulis mengembangkan *website* Notares, yang dirancang untuk membantu pengguna dalam mengelola keuangan mereka secara efektif. Aplikasi ini bukan hanya alat pengelolaan keuangan biasa, tetapi juga menyediakan fitur yang memberikan rekomendasi investasi dan tabungan, sesuai dengan profil risiko dan tujuan keuangan pengguna. Salah satu keunggulan utama dari aplikasi Notares adalah kemudahannya dalam pengoperasian, yang sangat sesuai dengan kemampuan digital mahasiswa masa kini. Aplikasi ini memungkinkan pengguna untuk melacak pengeluaran dan pendapatan mereka, merencanakan anggaran, serta mengevaluasi peluang investasi yang cocok dengan keadaan keuangan mereka. Dengan mengintegrasikan algoritma cerdas, Notares dapat memberikan rekomendasi yang disesuaikan, yang membantu mahasiswa membuat keputusan keuangan yang lebih baik.

Selain itu, Notares juga didesain dengan antarmuka yang ramah pengguna, yang memungkinkan navigasi yang intuitif dan akses cepat ke semua fitur. Kendati demikian, kami terus berusaha untuk meningkatkan aplikasi ini, dengan menambahkan fitur-fitur yang lebih inovatif dan menggunakan algoritma paling efektif untuk meningkatkan akurasi rekomendasi keuangan. Dengan fokus pada kebutuhan keuangan mahasiswa dan pemanfaatan teknologi terkini, Notares bertujuan untuk menjadi pendamping keuangan yang tidak hanya praktis tetapi juga edukatif, sehingga membantu generasi muda mengembangkan kebiasaan keuangan yang sehat untuk masa depan mereka.

Tim pengembang menggunakan Metode *Agile* dan Metode *Scrum* pada proyek ini. Metode *Agile* memprioritaskan respons terhadap perubahan selama siklus pengembangan daripada mengikuti rencana yang ketat, dengan tujuan utama mencapai kepuasan pelanggan. *Feature Driven Development* (FDD) adalah

salah satu metode *Agile* yang menggunakan iterasi pendek bertahap untuk menghasilkan perangkat lunak fungsional. FDD mengelola pengembangan perangkat lunak berdasarkan daftar fitur yang dibutuhkan oleh bisnis dan sangat adaptif terhadap perubahan persyaratan yang terlambat. Fokus utama FDD adalah menghasilkan output berkualitas tinggi di setiap tahap pengembangan [3].

Metode *Scrum* membagi tim besar menjadi beberapa tim kecil, di mana setiap tim kecil fokus pada tugas tertentu dalam proyek. Setiap tim harus mengelola hubungan antara tugas mereka dengan tugas tim lain dan mengadakan pertemuan *sprint* menggunakan *Scrum*. *Scrum* menciptakan lingkungan kerja yang kreatif dan produktif, memberikan kebebasan kepada pengembang untuk menentukan cara mereka bekerja. Anggota tim pengembang perangkat lunak dapat memilih tugas yang ingin mereka kerjakan, sambil tetap berkoordinasi dan bekerja secara lintas fungsi. Tim Notares membagikan tugas pembuatan fitur sesuai kemampuan anggotanya masing-masing, sesuai prinsip kebebasan pada Metode *Scrum* [4].

Oleh sebab itu, notares diperlukan karena sebagian besar aplikasi manajemen keuangan yang tersedia saat ini belum secara khusus dikembangkan untuk menjawab kebutuhan mahasiswa, baik dari segi fitur maupun pendekatan pengembangannya. Aplikasi sejenis cenderung bersifat generik, kurang mempertimbangkan keterbatasan anggaran, pola pengeluaran unik, serta kebutuhan edukasi finansial bagi mahasiswa. Selain itu, pendekatan pengembangan tradisional yang digunakan dalam aplikasi-aplikasi tersebut sering kali kurang fleksibel dalam menyesuaikan fitur berdasarkan umpan balik pengguna secara cepat. Oleh karena itu, penelitian ini menghadirkan Notares dengan keunggulan utama pada proses pengembangannya yang menggunakan metode *Agile* dengan kerangka kerja *Scrum*. Pendekatan ini memungkinkan iterasi cepat, peningkatan fitur yang berkelanjutan, serta integrasi masukan pengguna secara langsung dalam setiap siklus *sprint*. Dengan demikian, *Scrum* tidak hanya digunakan sebagai metode manajemen proyek, tetapi juga sebagai strategi inovatif untuk menghasilkan solusi keuangan digital yang lebih relevan, adaptif, dan tepat guna bagi mahasiswa.

2. Tinjauan Pustaka/Penelitian Sebelumnya

2.1 Penelitian Terdahulu yang Terkait

Metode *Scrum* umumnya memungkinkan tim untuk merespons perubahan kebutuhan dan prioritas, serta meningkatkan kolaborasi dan transparansi dalam proses pengembangan. Proses *Scrum* dapat diterapkan pada pengembangan *website* di berbagai bidang. Salah satu contoh adalah proyek *ListeningTime* yang dikembangkan oleh Noordman et al [5]. pada tahun 2017. Proyek ini adalah website kesehatan yang menggunakan metode *Scrum* untuk mengatasi masalah koordinasi dan komunikasi antara pasien yang sakit parah dengan tenaga kesehatan yang sangat sibuk. Dengan metode *Scrum*, proyek ini berhasil mengurangi kesulitan komunikasi yang disebabkan oleh keterbatasan waktu pasien dan padatnya jadwal tenaga kesehatan. Pengalaman dari proyek *ListeningTime* memberikan wawasan penting tentang bagaimana *Scrum* dapat meningkatkan koordinasi dalam lingkungan yang sangat dinamis, menjadi acuan pertama bagi proyek di studi ini. Contoh lain penerapan metode *agile* untuk pengembangan E-Health untuk pendidikan pasien [6].

Melanjutkan dari contoh di bidang kesehatan, Sasmito et al [7]. pada tahun 2020 mengembangkan proyek aplikasi pemetaan telur asin berbasis web dengan menggunakan metode *Scrum*. Dalam proyek ini, metode *Scrum* digunakan untuk mengatasi masalah visualisasi data produksi telur asin yang kurang menarik dan penggunaan *hyperlink* yang belum optimal di *website* tersebut. Metode *Scrum* membantu tim untuk mengelola dan mengembangkan fitur visualisasi yang lebih menarik dan penggunaan *hyperlink* yang lebih efektif. Pengalaman dari proyek ini menunjukkan fleksibilitas *Scrum* dalam memperbaiki aspek visual dan navigasi situs web, dan menjadi acuan kedua bagi proyek di studi ini.

Selain itu, dalam bidang pendidikan, Laila et al. [8] pada tahun 2021 mengembangkan proyek Sistem ADI STETSA SMAN 4 Malang, sebuah aplikasi bimbingan konseling berbasis *web*. Proyek ini menggunakan metode *Scrum* untuk mengatasi masalah alur pelayanan yang belum jelas. Dengan menerapkan *Scrum*, tim berhasil menciptakan alur pelayanan yang lebih terstruktur dan efisien dalam aplikasi bimbingan konseling ini. Contoh ini menunjukkan bagaimana *Scrum* dapat diterapkan untuk memperbaiki proses layanan internal sekolah, menjadi acuan ketiga bagi proyek di studi ini. Contoh lainnya, model *Agile Scrum* untuk Pengembangan Sistem Informasi Penerimaan Mahasiswa Baru Berbasis Web di RS Sint Carolus Jakarta [9]

Di sektor ekonomi, Alim et al. [10] pada tahun 2023 mengembangkan Aplikasi Cangkingan, sebuah *marketplace* berbasis *web* untuk membantu UMKM di Kabupaten Banyumas. Aplikasi ini dirancang untuk mengelola inventaris, memproses pesanan, dan memasarkan produk kuliner secara *online*. Metode *Scrum* membantu dalam pengumpulan dan pengelolaan kebutuhan pembangunan sistem dengan membagi kebutuhan tersebut menjadi bagian-bagian kecil yang dapat dikelola dan dikembangkan dalam *beberapa sprint*. Proyek ini menjadi contoh bagaimana *Scrum* dapat digunakan untuk meningkatkan efisiensi operasional UMKM, menjadi acuan keempat bagi proyek penulis. Demikian juga di sektor pengembangan aplikasi pariwisata berbasis mobil, dengan menerapkan metodologi *Scrum*, proyek aplikasi seluler dikonseptualisasikan untuk memperkuat visibilitas penawaran wisata [11].

Terakhir, studi tentang pengembangan aplikasi *mobile* untuk pemantauan proyek di PT GSP, menggantikan metode manual yang kurang efisien. Hasilnya, aplikasi meningkatkan efektivitas proyek dengan 100% keberhasilan pada pengujian *black-box* dan 74% kepuasan pengguna dalam UAT. Aplikasi ini kini digunakan untuk semua proyek, dengan saran pelatihan bagi pengguna baru [12]. Pengalaman dari proyek ini menunjukkan bagaimana *Scrum* dapat membantu organisasi dalam mengelola dan memvisualisasikan data secara efektif, menjadi acuan kelima bagi proyek penulis.

Dengan mengambil setiap laporan penggunaan metode *Scrum* dari berbagai proyek di bidang kesehatan, pendidikan, ekonomi, dan bisnis, penulis berharap dapat mengimplementasikan metode *Scrum* secara efektif dalam pengembangan proyek Notares.

2.2 Scrum

Scrum adalah metodologi pengembangan perangkat lunak yang mengedepankan proses iteratif dan inkremental, serta menekankan pentingnya transparansi, pemeriksaan, dan adaptasi dalam pekerjaan tim. Metode ini mengatur tim kecil yang terdiri dari Pemilik Produk, *Scrum Master*, dan Pengembang untuk bekerja dalam siklus waktu yang disebut *sprint*, dengan tujuan mencapai target produk secara efektif. Tim *Scrum* mengikuti serangkaian acara yang terjadwal seperti perencanaan *sprint*, *Scrum* harian, tinjauan *sprint*, dan retrospeksi *sprint*, yang semuanya mendukung tim dalam beradaptasi dengan perubahan selama proyek berlangsung. Nilai-nilai inti seperti komitmen, fokus, keterbukaan, rasa hormat, dan keberanian juga sangat penting, karena nilai-nilai ini membimbing perilaku dan interaksi tim selama proyek [13].

2.3 Website Development dengan NextJS

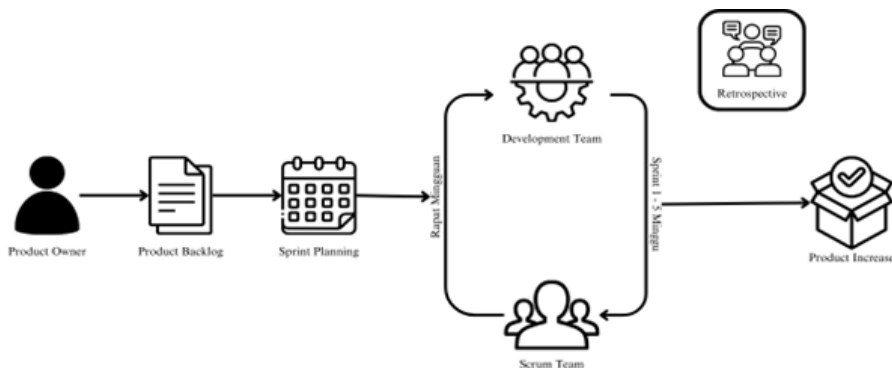
Next.js merupakan *framework* yang berbasis *Javascript*. Penggunaan *framework* *Next.js* ini sebagai inti dalam pengembangan *Website* Notares menjadi lebih responsif, cepat, dan fleksibel dalam proses mengakhiri untuk aplikasi *web*. Selain itu, *framework* *Next.js* memudahkan pengembang untuk menambahkan fungsionalitas tambahan jika diperlukan, karena struktur penulisan kode sudah mengikuti aturan yang memungkinkan penggunaan ulang. Fungsi dan fitur yang ada sudah sesuai dengan aturan terkait komponen, tata letak, dan halaman [14].

Pemanfaatan *NextJS* dan penerapannya telah memberikan kontribusi besar dalam meningkatkan performa, optimasi mesin pencari, dan fungsi situs web. Teknologi ini memungkinkan tim pengembang untuk

menyediakan pengalaman pengguna yang unggul serta memastikan visibilitas dan aksesibilitas situs bagi pengguna maupun mesin pencari [15].

Seiring dengan kemajuan pengembangan web, mengikuti tren ini menjadi sangat penting bagi pengembang dan bisnis yang ingin membangun aplikasi yang inovatif, efisien, dan ramah pengguna. Pemilihan teknologi harus sesuai dengan persyaratan proyek, kebutuhan skalabilitas, dan harapan pengguna yang terus berubah di dunia digital [16].

3. Metodologi



Gambar 1 Fase-fase Scrum dalam Pengembangan Website Notares

Dalam proyek pengembangan *website* Notares, kerangka kerja *Scrum* dipilih sebagai metode utama. *Scrum*, yang dikenal dengan pendekatan yang fleksibel dan ringan dalam pengembangan perangkat lunak, memberikan kerangka kerja yang jelas melalui serangkaian aturan dan tugas yang ditujukan untuk mengoptimalkan kerja tim. Rincian tahapan pengembangan *Website* Notares menggunakan metode *Scrum*, termasuk pengumpulan *User Stories*, pembuatan *Product Backlog*, perencanaan *Sprint*, dan pengujian fitur yang telah dikembangkan. Gambar 1 menunjukkan gambaran umum tentang bagaimana metode *Scrum* diaplikasikan dalam pengembangan *website* Notares. Penjelasan dari fase-fase *Scrum* dalam pengembangan *Website* Notares seperti yang ditunjukkan di gambar, mencakup lima tahapan berikut:

3.1 Pengumpulan *User Stories*

Tahap awal dalam pengembangan *Website* Notares adalah pengumpulan *User Stories* guna mengetahui perspektif kebutuhan pengguna sehingga pengembangan *website* didasari oleh proses untuk menyelesaikan kebutuhan tersebut.

3.2 Pembuatan *Product Backlog*

Product Backlog berisi daftar tugas, *activity flow*, *use case diagram*, dan *Entity Relationship Diagram* (ERD) yang berasal dari *User Stories* dan studi literatur terkait.

3.3 Perencanaan *Sprint*

Perencanaan *Sprint* dalam pengembangan *Website* Notares dibagi menjadi dua bagian. Bagian pertama adalah *Sprint Planning* untuk mengevaluasi daftar tugas yang ada pada *Product Backlog*. Sedangkan, bagian kedua adalah *Sprint Backlog* yang terdiri dari detail tugas-tugas yang harus diselesaikan oleh anggota tim selama periode *Sprint*.

3.4 Pelaksanaan Sprint

Sepanjang pelaksanaan *Sprint*, tim mengadakan pertemuan harian yang dikenal sebagai *Daily Scrum* untuk menyelaraskan pekerjaan yang dilakukan oleh para pengembang, memastikan kemajuan sejalan dengan rencana, dan mengidentifikasi hambatan yang perlu ditangani.

3.5 Evaluasi Sprint

Evaluasi *Sprint* dilakukan untuk mengevaluasi kemajuan pengembangan *Website* Notares. Pengujian juga dilakukan untuk memverifikasi apakah fitur-fitur yang dikembangkan dalam *Sprint* sebelumnya berfungsi dengan benar. Setiap kesalahan yang ditemukan diperbaiki dan diperbaiki lebih lanjut dalam *Sprint* berikutnya untuk mencapai hasil akhir yang baik.

4. Hasil dan Pembahasan

Dalam pengembangan *Website* Notares, penulis menggunakan *User Stories* untuk mengidentifikasi dan mendokumentasikan kebutuhan pengguna, khususnya mahasiswa Sistem Informasi ITS, dalam manajemen keuangan. Melalui wawancara intensif, penulis mengumpulkan berbagai kebutuhan seperti melacak pengeluaran, membuat anggaran, dan menerima rekomendasi investasi. Setiap kebutuhan ini kemudian dirinci dalam Tabel 1, yang menjadi dasar pengembangan aplikasi.

Backlog produk terdiri dari fitur penting yang diperlukan untuk pengoperasian *website*, termasuk pendaftaran pengguna, operasi *Create Read Update & Delete* (CRUD) pada catatan keuangan, dan visualisasi data melalui grafik dan tabel. Penulis juga mempersiapkan *Diagram Use Case* untuk menggambarkan secara visual interaksi pengguna dengan sistem, serta *Entity Relationship Diagram* (ERD) yang memastikan efisiensi struktur basis data. Fitur-fitur ini dirancang untuk memberikan pengalaman pengguna yang intuitif dan memudahkan navigasi dalam aplikasi.

Proses pengembangan dibagi dalam lima *sprint*, dengan setiap *sprint* diorganisir selama sesi *Sprint Planning* yang melibatkan seluruh tim *Scrum*. Dalam sesi ini, penulis mengevaluasi dan memprioritaskan fitur berdasarkan urgensi dan sumber daya yang diperlukan. Beberapa *sprint* ini diatur untuk memastikan alokasi sumber daya yang efisien dan pemenuhan tenggat waktu pengembangan secara efektif, untuk menghasilkan platform yang tidak hanya memenuhi kebutuhan pengelolaan keuangan mahasiswa tetapi juga mudah digunakan.

4.1 User Stories

User Stories adalah metode untuk menggambarkan kebutuhan atau fitur dari sudut pandang pengguna dalam pengembangan perangkat lunak. Hal ini sangat penting karena daftar *User Stories* menjadi dasar bagi proses pengembangan aplikasi selanjutnya. Berdasarkan wawancara mengenai kebutuhan mahasiswa Sistem Informasi ITS mengenai manajemen keuangan diperoleh berbagai *User Stories* sebagaimana yang terdapat dalam Tabel 1.

4.2 Product Backlog

Backlog produk adalah daftar prioritas dari semua fitur, fungsi, peningkatan, dan perbaikan yang diperlukan untuk pengembangan sebuah proyek. Dalam konteks pengembangan *website* Notares, *backlog* produk mencakup berbagai elemen penting yang harus diselesaikan oleh tim pengembang. Setiap item dalam *backlog* produk diberi prioritas berdasarkan kebutuhan dan pentingnya bagi pengguna akhir. Tabel 2 menampilkan contoh item-item yang terdapat dalam *backlog* produk untuk *website* Notares beserta prioritasnya.

Tabel 1. Tabel software dan hardware pendukung [Times New Roman 8 normal]

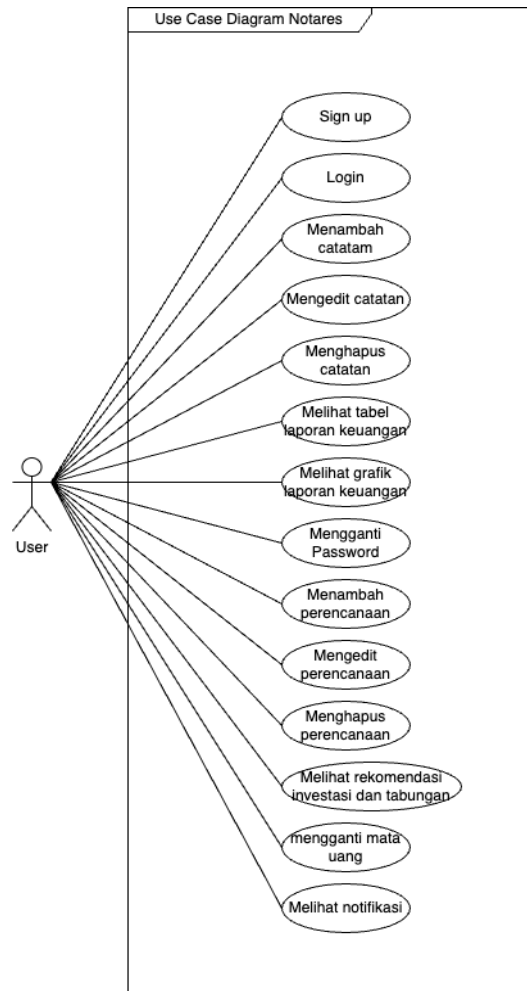
Hasil yang Diharapkan	Kesimpulan
Memiliki fitur manajemen keuangan yang lengkap meliputi penambahan, pengeditan, hingga penghapusan catatan keuangan serta detail-detail lainnya, seperti, hari, deskripsi, dan kategori catatan.	Memastikan operasi CRUD dalam pencatatan keuangan dan fitur tanggal, kategori, deskripsi, pemasukan, pengeluaran, dan saldo untuk catatan keuangan dapat digunakan oleh pemakai.
Memiliki <i>dashboard</i> yang menampilkan laporan keuangan secara grafik.	Memastikan keterhubungan basis data dengan <i>dashboard</i> dalam bentuk grafik laporan keuangan
Memiliki <i>dashboard</i> yang menampilkan saldo dan total pengeluaran selama periode yang ditentukan.	Memastikan keterhubungan basis data dengan <i>dashboard</i> yang memiliki fitur filter data berdasarkan periode laporan yang dipilih pemakai.
Memiliki sistem manajemen akun yang dapat dikustomisasi serta aman.	Memastikan keamanan dan kenyamanan personalisasi akun.
Memiliki antarmuka yang ramah serta lancar.	Memastikan <i>User Interface</i> (UI) yang ramah pengguna serta <i>hosting</i> dan arsitektur basis data yang baik.
Memiliki fitur untuk perencanaan keuangan meliputi biaya hidup hingga target.	Memastikan berjalannya fitur perencanaan keuangan dengan personalisasi mengikuti kebutuhan keuangan pengguna.
Memiliki rekomendasi investasi dan tabungan.	Memastikan algoritma mengenai rekomendasi investasi berdasarkan penghasilan, kebutuhan, dan preferensi risiko investasi.
Memiliki <i>multi currency</i> yang dapat digunakan.	Memastikan sistem menyediakan berbagai mata uang serta konversinya secara akurat dan <i>real time</i> .
Memiliki notifikasi apabila pengeluaran tidak normal dan melewati batas perencanaan pengeluaran.	Memastikan sistem memiliki algoritma yang dapat memahami kebiasaan pengguna dengan <i>output</i> notifikasi ketidaknormalan pengeluaran.
Memiliki notifikasi pembayaran <i>subscription</i> atau layanan yang mengharuskan pembayaran bulanan.	Memastikan sistem dapat merekam tanggal tenggat pembayaran serta memberikan <i>output</i> berupa notifikasi apabila tenggat pembayaran sudah dekat.

Tabel 2. *Product Backlog*

Item	Prioritas
Desain basis data	Tinggi
Halaman <i>Signup</i> /Pendaftaran User	Tinggi
Halaman Masuk/ <i>Login</i>	Tinggi
<i>View</i> Tabel Data	Sedang
<i>Chart</i> Catatan Keuangan	Sedang
Profil User	Sedang
Operasi CRUD dalam Pencatatan Keuangan	Sedang
Desain Antarmuka Pengguna	Rendah
Halaman Perencanaan Biaya	Rendah
Halaman Rekomendasi Investasi dan Tabungan	Rendah
Pilihan <i>Multi Currency</i>	Rendah
Algoritma Pengeluaran Tidak Normal	Rendah
Notifikasi Tenggat Pembayaran	Rendah

4.2.1 Use Case Diagram

Use case diagram adalah representasi visual dari interaksi antara pengguna dan sistem yang dikembangkan. *Use Case Diagram* untuk *website* Notares menggambarkan berbagai interaksi yang mungkin dilakukan oleh pengguna dengan sistem. Diagram ini menunjukkan hubungan antara pengguna dan fitur-fitur utama yang tersedia dalam *website* Notares, termasuk *Sign Up*, *Login*, Menambah catatan, Mengedit catatan, Menghapus catatan, Melihat tabel laporan keuangan, Melihat grafik laporan keuangan, dan Mengganti password. Gambar berikut ini adalah ilustrasi dari *Use Case Diagram* untuk *website* Notares:

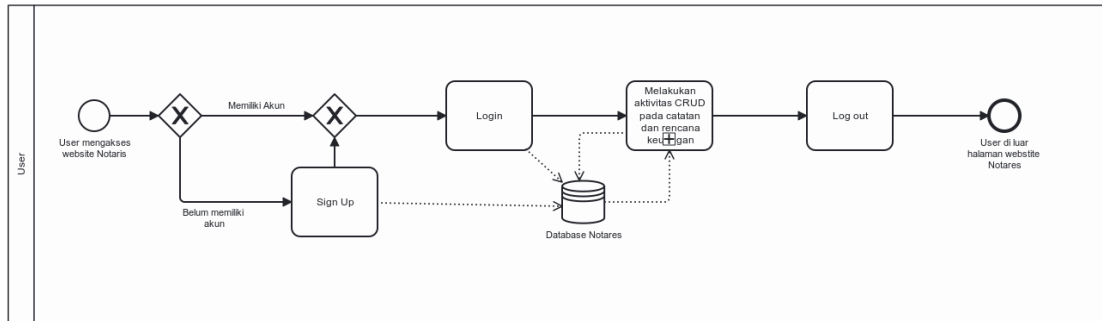


Gambar 2. Use case diagram website Notares

Use case diagram menjelaskan fitur yang dapat dilakukan pengguna di *website* Notares meliputi *sign up*, *login*, CRUD di catatan keuangan dan perencanaan keuangan, melihat grafik dari catatan, melihat rekomendasi keuangan, hingga fitur-fitur personalisasi.

4.2.2 Activity Flow

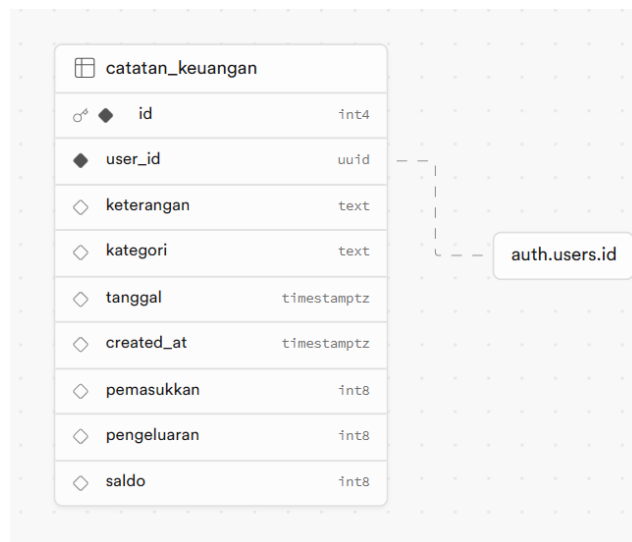
Activity flow adalah representasi diagramatik dari alur kerja atau proses yang ada dalam sistem. Ini menggambarkan langkah-langkah yang harus diambil untuk menyelesaikan suatu tugas atau aktivitas dari awal hingga akhir. Dalam pengembangan *website* Notares, *activity flow* membantu dalam memahami bagaimana data dan pengguna berpindah melalui sistem untuk menyelesaikan berbagai tugas. *Activity Flow* untuk *website* Notares menggambarkan urutan aktivitas yang terjadi dalam penggunaan *website*, mulai dari saat pengguna mendaftar, masuk, hingga melakukan operasi CRUD pada catatan dan perencanaan keuangan. Diagram ini membantu dalam memvisualisasikan proses alur kerja dari sudut pandang pengguna. Gambar 3 adalah ilustrasi dari *Activity Flow* untuk *website* Notares:



Gambar 3. Activity Flow Website Notares

4.2.3 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan diagram yang dibuat untuk mempermudah dalam membuat basis data. Dengan ERD, relasi antar entitas yang ada di dalam basis data dapat dilihat dan diketahui dengan jelas. Hal ini sangat penting dalam proses pengembangan sehingga proses penambahan, pengubahan, dan penghapusan catatan dalam *website* ini dapat berjalan dengan baik. Gambar 4 menunjukkan ERD publik sederhana yang digunakan dalam pengembangan Notares:



Gambar 4. ERD Publik Website Notares

4.3 Sprint

Pada fase ini dibagi menjadi dua poin, yaitu, fase *sprint planning* dan *sprint backlog*. Selama perencanaan *Sprint*, tim *Scrum* mengadakan pertemuan untuk mengevaluasi daftar yang ada di *Product Backlog*. Tim *Scrum* membahas setiap fitur yang akan dikembangkan oleh masing-masing anggota tim serta memperkirakan waktu yang dibutuhkan untuk penyelesaiannya.

4.3.1 Sprint Planning

Sprint Planning dirancang dengan membagi pekerjaan menjadi lima *sprint* yang berbeda. Pembagian ini didasarkan pada urgensi pekerjaan atau fitur tersebut dari kebutuhan pelanggan serta kesesuaian teori

dalam pengembangan perangkat lunak dari sisi *developer* atau IT. Tabel 3 sampai Tabel 7 adalah tabel *Sprint Planning* dari rencana pengembangan *website* Notares:

Tabel 2. *Sprint Planning 1*

<i>Sprint Goal</i>	Mengembangkan fitur dasar untuk <i>website</i> Notares yang meliputi desain basis data, sistem pendaftaran user, dan sistem login.
<i>Sprint Duration</i>	5 hari
<i>Team Members</i>	Farhan Adika Suwardana, Rahmat Ramadhan Permana, Agung Budi Prasetya, Arayzi Rayyansyah, Arjuna Putra Kharisma

Tabel 3. *Sprint Planning 2*

<i>Sprint Goal</i>	Mengembangkan fitur lanjutan untuk <i>website</i> Notares yang meliputi view tabel data, <i>chart</i> catatan keuangan, serta CRUD dalam catatan keuangan.
<i>Sprint Duration</i>	6 hari
<i>Team Members</i>	Farhan Adika Suwardana, Rahmat Ramadhan Permana, Agung Budi Prasetya, Arayzi Rayyansyah, Arjuna Putra Kharisma

Tabel 4. *Sprint Planning 3*

<i>Sprint Goal</i>	Mengembangkan finishing untuk <i>website</i> Notares yang meliputi profil <i>user</i> dan penyelesaian desain antarmuka.
<i>Sprint Duration</i>	5 hari
<i>Team Members</i>	Farhan Adika Suwardana, Rahmat Ramadhan Permana, Agung Budi Prasetya, Arayzi Rayyansyah, Arjuna Putra Kharisma

Tabel 5. *Sprint Planning 4*

<i>Sprint Goal</i>	Mengembangkan fitur tambahan untuk <i>website</i> Notares yang meliputi implementasi desain antarmuka, perencanaan keuangan, rekomendasi investasi atau tabungan.
<i>Sprint Duration</i>	6 hari
<i>Team Members</i>	Farhan Adika Suwardana, Rahmat Ramadhan Permana, Agung Budi Prasetya, Arayzi Rayyansyah, Arjuna Putra Kharisma

Tabel 6. *Sprint Planning 5*

<i>Sprint Goal</i>	Mengembangkan fitur tambahan untuk <i>website</i> Notares yang meliputi notifikasi apabila pengeluaran tidak normal, notifikasi pembayaran <i>subscription</i> , serta pilihan mengganti mata uang.
<i>Sprint Duration</i>	6 hari
<i>Team Members</i>	Farhan Adika Suwardana, Rahmat Ramadhan Permana, Agung Budi Prasetya, Arayzi Rayyansyah, Arjuna Putra Kharisma

4.3.2 *Sprint Backlog*

Sprint Backlog dirancang untuk mengestimasi dari setiap *sprint planning* terhadap waktu yang dibutuhkan dalam setiap fasenya. *Sprint backlog* direncanakan berdasarkan pengamatan kami terhadap literatur terdahulu mengenai *scrum* pada *development* serta pustaka yang terkait dengan pengembangan *website* dengan *framework react*. Tabel 8 sampai Tabel 11 menampilkan *Sprint Backlog* dari *website* Notares:

Tabel 7. *Sprint Backlog 1*

<i>Sprint</i>	<i>Task</i>	Estimasi Waktu (Hari/Jam)				
		1	2	3	4	5
Pengembangan desain <i>database</i> , sistem pendaftaran, dan login.	Desain <i>database</i>	3				
	Sistem pendaftaran <i>user</i>		3	1		
	Sistem <i>login</i>		2	1		
	Implementasi <i>database</i> pada sistem			5	4	
	<i>Testing</i>			1	2	1
Total		23 Jam				

Tabel 8. *Sprint Backlog 2*

<i>Sprint</i>	<i>Task</i>	Estimasi Waktu (Hari/Jam)					
		1	2	3	4	5	6
Pengembangan sistem <i>view</i> tabel data, <i>chart</i> catatan keuangan, serta CRUD dalam catatan keuangan.	Desain sistem CRUD	4					
	Pengembangan halaman catatan keuangan	2	3				
	Pengembangan tabel keuangan	1	2	3			
	Pengembangan sistem CRUD			3	4	5	
	<i>Testing</i>		1			2	2
Total		32 Jam					

Tabel 9. *Sprint Backlog 3*

<i>Sprint</i>	<i>Task</i>	Estimasi Waktu (Hari/Jam)				
		1	2	3	4	5
Pengembangan desain antarmuka, implementasi pada <i>website</i> dan kustomisasi profil user	Desain antarmuka	6				
	Implementasi desain pada halaman <i>website</i>		3	3		
	Pengembangan kustomisasi profil user		2	2	4	
	<i>testing</i>			1		3
Total		24 Jam				

Tabel 10. *Sprint Backlog 4*

<i>Sprint</i>	<i>Task</i>	Estimasi Waktu (Hari/Jam)					
		1	2	3	4	5	6
Pengembangan implementasi desain antarmuka, perencanaan keuangan, dan rekomendasi investasi atau tabungan	Implementasi desain antarmuka ke <i>website Notares</i>	6					
	Desain alur sistem perencanaan dan rekomendasi keuangan	4					
	Desain antarmuka dari sistem perencanaan dan rekomendasi keuangan			2			
	Penyesuaian database			4			
	Pengembangan model <i>Artificial Intelligence</i>			2	2	1	
	Pengembangan sistem					5	5
	<i>Testing</i>			2	2		2
Total		40 Jam					

Tabel 11. *Sprint Backlog 5*

Sprint	Task	Estimasi Waktu (Hari/Jam)					
		1	2	3	4	5	6
	Desain alur notifikasi	3					
Pengembangan notifikasi apabila pengeluaran tidak normal, pembayaran <i>subscription</i> , serta pilihan mengganti mata uang	Desain antarmuka notifikasi	2	2				
	Penyesuaian <i>database</i>	1	3		3		
	Pengembangan fitur mengganti mata uang		3	1			
	Pengembangan sistem notifikasi			4	4	4	1
	<i>Testing</i>		2			2	2
Total		37 Jam					

4.4 Hasil dan Evaluasi Sprint

Sprint yang meliputi *Sprint Planning* dan *Sprint Backlog* dijalankan dengan rencana serta estimasi waktu yang ditentukan. Evaluasi diperlukan di setiap *sprint* agar tidak menjadi masalah yang berkelanjutan di sesi *sprint* berikutnya. Secara keseluruhan Tabel 12 menampilkan hasil *Sprint*

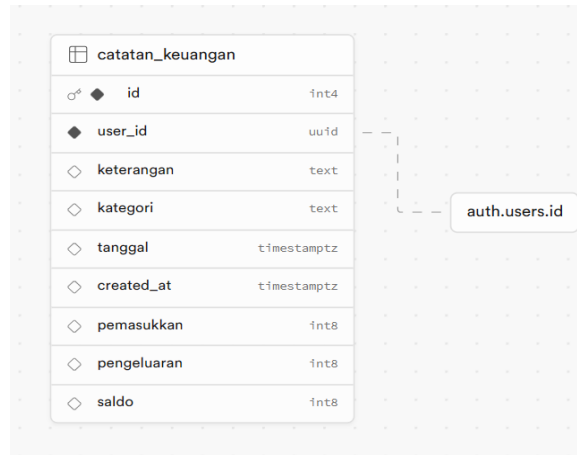
Tabel 12. Hasil *Sprint*

<i>Sprint</i>	<i>Goal</i>	Status
<i>Sprint 1</i>	Mengembangkan fitur dasar untuk <i>website</i> Notares yang meliputi desain <i>database</i> , sistem pendaftaran <i>user</i> , dan sistem <i>login</i> .	Sukses
<i>Sprint 2</i>	Mengembangkan fitur lanjutan untuk <i>website</i> Notares yang meliputi <i>view</i> tabel data, chart catatan keuangan, serta CRUD dalam catatan keuangan.	Sukses
<i>Sprint 3</i>	Mengembangkan <i>finishing</i> untuk <i>website</i> Notares yang meliputi profil user dan penyelesaian desain antarmuka.	Sukses
<i>Sprint 4</i>	Mengembangkan fitur tambahan untuk <i>website</i> Notares yang meliputi implementasi desain antarmuka, perencanaan keuangan, rekomendasi investasi atau tabungan.	Belum
<i>Sprint 5</i>	Mengembangkan fitur tambahan untuk <i>website</i> Notares yang meliputi notifikasi apabila pengeluaran tidak normal, notifikasi pembayaran <i>subscription</i> , serta pilihan mengganti mata uang.	Belum

Berikut adalah hasil serta evaluasi dari masing-masing *sprint*.

4.4.1 *Sprint 1*

Goal dari *sprint* yang pertama adalah mengembangkan fitur dasar untuk *website* Notares yang meliputi desain *database*, sistem pendaftaran *user*, dan sistem *login*. Berikut adalah gambar hasil dari pelaksanaan *sprint* pertama.



Gambar 5. ERD Publik *Website Notares*

Gambar 5 adalah ERD dari desain *database* menggunakan supabase untuk *website* Notares. Dalam gambar tersebut *database* digunakan untuk memenuhi fitur dasar Notares yang meliputi sistem pendaftaran dan CRUD untuk catatan keuangan. Selanjutnya, Gambar 6 dan Gambar 7 merupakan gambar dari halaman pendaftaran dan login dari *website* Notares.

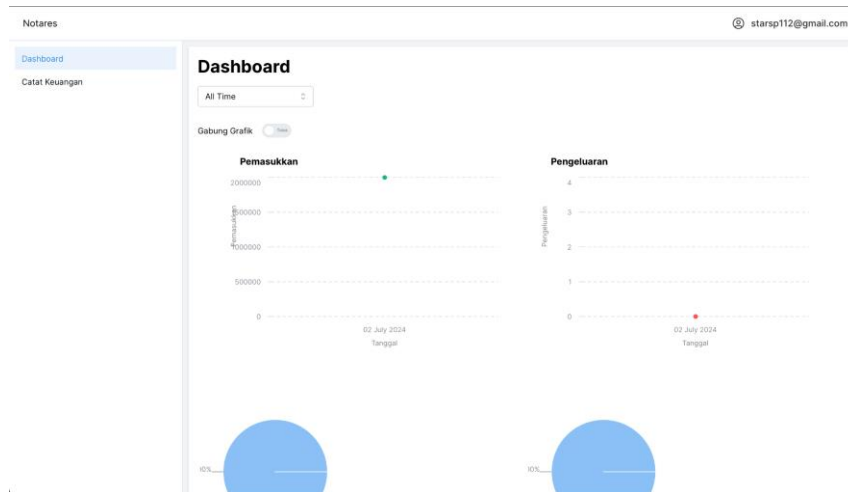
Gambar 6. Halaman *login Website Notares*

Gambar 7. Halaman pendaftaran *user website Notares*

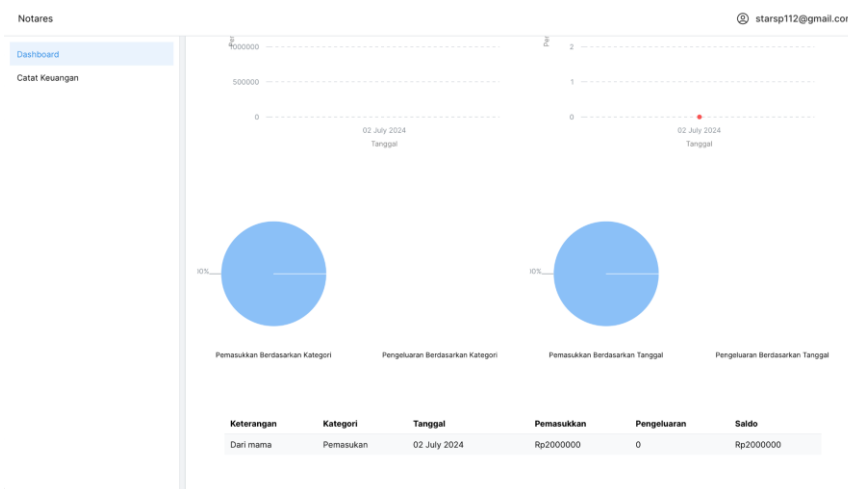
Kedua gambar di atas menampilkan halaman *login* dan halaman pendaftaran user. Halaman *login* merupakan halaman pertama yang muncul pada saat *website* Notares dikunjungi. *User* diminta untuk memasukkan email serta sandi untuk melanjutkan ke halaman beranda, tetapi apabila *user* belum memiliki akun maka mereka dapat memilih daftar. Nama depan, nama belakang, email, serta sandi perlu di-*input* *user* untuk mendaftarkan akun mereka di halaman pendaftaran *user*. Evaluasi untuk *sprint* pertama ini, yakni, beberapa tugas dilakukan melebihi estimasi pengerjaan serta beberapa yang lain lebih cepat dari estimasi. *Sprint* selanjutnya, kita memastikan estimasi waktu sesuai dengan kebutuhan sebenarnya, kami merubah detail dari *sprint backlog* kedua dan seterusnya berdasarkan pengalaman kami di *sprint* pertama serta tinjauan pustaka yang ada.

4.4.2 Sprint 2

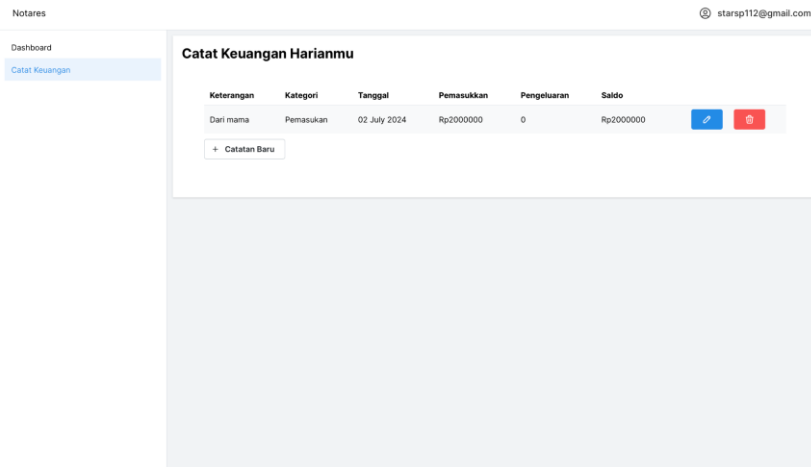
Goal dari *sprint* kedua adalah mengembangkan fitur lanjutan untuk *website* Notares yang meliputi *view* tabel data, *chart* catatan keuangan, serta *CRUD* dalam catatan keuangan. Berikut adalah gambar hasil dari pelaksanaan *sprint* kedua.



Gambar 8. Halaman *dashboard* website Notares



Gambar 9. Halaman *dashboard* website Notares (*chart*)

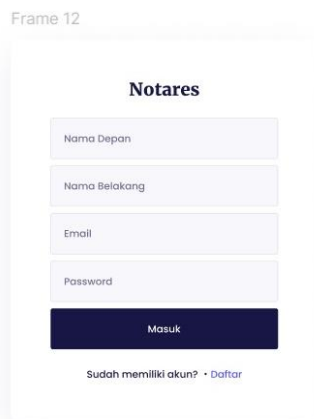


Gambar 10. Halaman catat keuangan *website Notares*

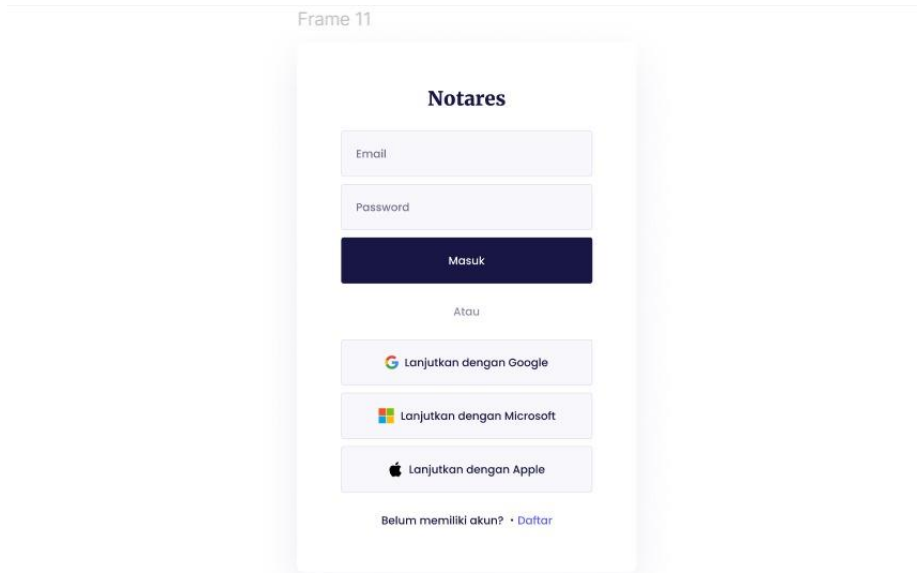
Gambar 8 dan Gambar 9 merupakan halaman *dashboard website Notares*, di halaman ini *user* dapat melihat rangkuman dari keuangan mereka. Terdapat *chart* garis dan lingkaran yang memberikan informasi keuangan dari *user*. Terdapat juga tabel catatan keuangan yang terdapat di bagian bawah. Sedangkan untuk Gambar 10 adalah halaman catat keuangan, di sini *user* dapat melakukan operasi CRUD di catatan mereka. Sel yang dapat diisi meliputi keterangan, kategori, tanggal, pengeluaran, serta pendapatan. Terdapat kolom saldo yang ditulis dengan mengambil data dari input *user*. Evaluasi dari *sprint* kedua ini, yakni, beban kerja salah seorang anggota lebih berat dari anggota lain, hal ini dikarenakan tugas pengembang (*coding*) sangat sentral di *sprint* kedua ini. *Sprint* selanjutnya, kami memastikan tidak adanya ketimpangan beban kerja antaranggota dengan memberikan tugas lebih detail berdasarkan pengalaman kami di kedua *sprint* pertama serta kajian pustaka yang telah dilakukan.

4.4.3 Sprint 3

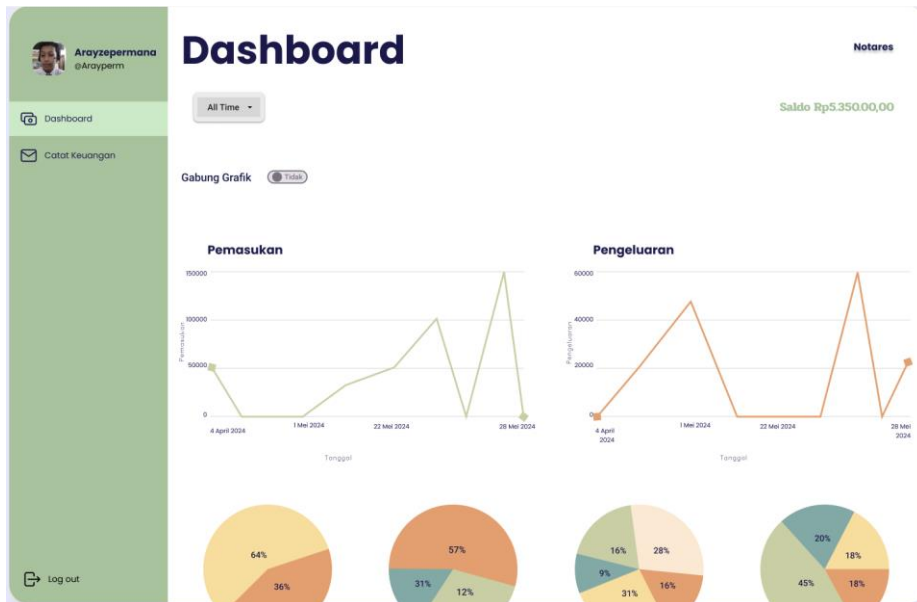
Goal dari *sprint* ketiga adalah mengembangkan finishing untuk *website Notares* yang meliputi profil *user* dan penyelesaian desain antarmuka. Gambar 11 dan Gambar 12 merupakan hasil dari pelaksanaan *sprint* ketiga.



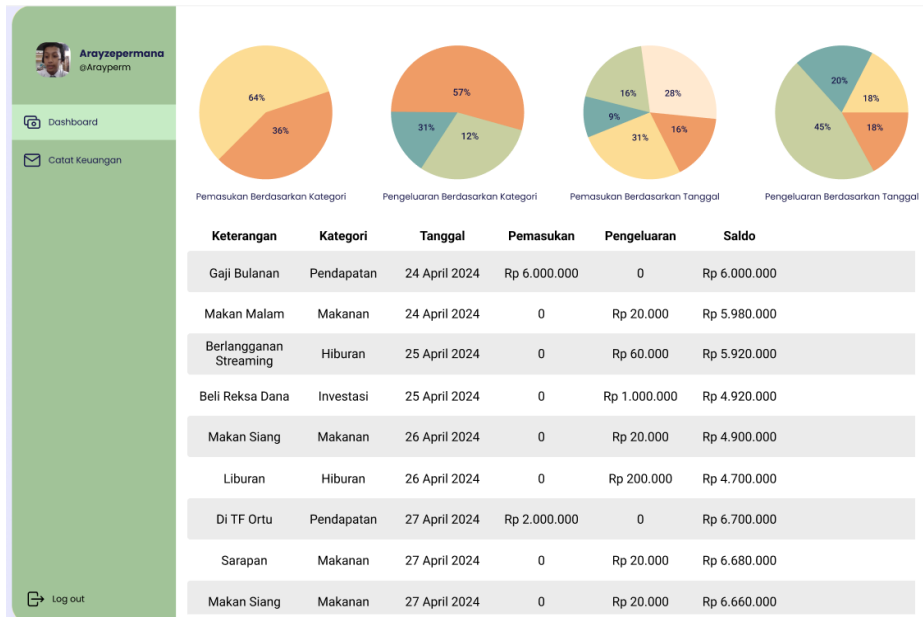
Gambar 11. Desain antarmuka halaman pendaftaran akun



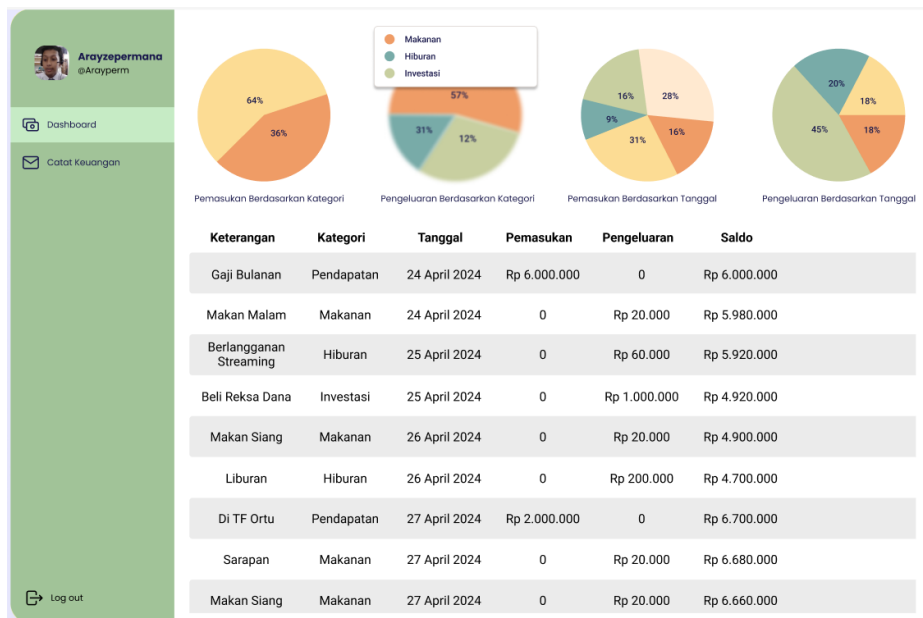
Gambar 12. Desain antarmuka halaman *login*



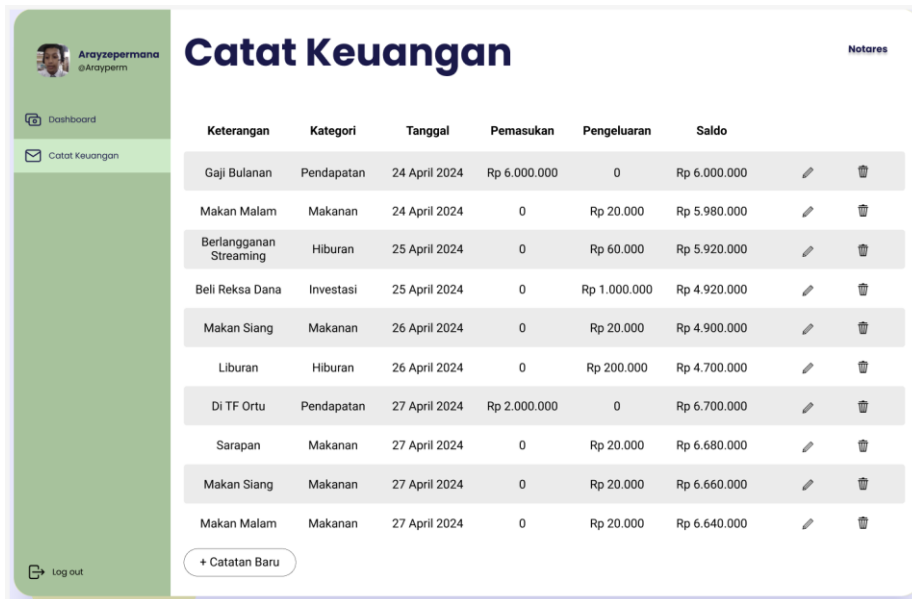
Gambar 13. Desain antarmuka halaman *dashboard* (line chart)



Gambar 14. Desain antarmuka halaman *dashboard (pie chart)* dan tabel



Gambar 15. Desain antarmuka halaman *dashboard (pie chart hover)* dan tabel



Gambar 16. Desain antarmuka halaman catat keuangan

Gambar 13 sampai Gambar 16 merupakan desain antarmuka yang nantinya diimplementasikan di *website* Notares di *sprint* keempat. Evaluasi pada *sprint* ini adalah kurangnya anggota tim yang paham mengenai teori desain. Langkah yang penulis ambil selanjutnya, yakni, memberi tambahan waktu di *sprint* keempat untuk sedikit merevisi desain yang ada di *sprint* ketiga.

Sprint keempat dan kelima memiliki status “belum”, hal ini dimaksudkan bahwa kedua *sprint* tersebut belum berjalan hingga laporan *paper* ini ditulis.

Pengembangan Notares menunjukkan hasil yang cukup signifikan dalam memenuhi kebutuhan mahasiswa terkait manajemen keuangan. Penerapan metode *Scrum* berperan penting dalam keberhasilan ini, dengan menyediakan kerangka kerja yang memungkinkan pembagian tugas yang jelas, iterasi yang terstruktur, serta evaluasi berkelanjutan terhadap progres dan kualitas fitur yang dikembangkan. Dari sisi kelebihan, metode *Scrum* mendukung kolaborasi tim yang erat, terlihat dari penyelesaian *sprint* secara bertahap dengan fokus yang terarah pada fitur utama. Fitur-fitur seperti pencatatan keuangan, visualisasi laporan, dan rekomendasi investasi berhasil dikembangkan dalam rentang waktu yang relatif singkat, yaitu total 79 jam kerja untuk menyelesaikan tiga dari lima *sprint* yang direncanakan. *Scrum* juga memungkinkan fleksibilitas dalam merespons masukan pengguna secara langsung, meningkatkan kualitas sistem melalui pengujian dan refleksi pada akhir tiap *sprint*.

Namun demikian, terdapat beberapa kelemahan dalam pelaksanaan proyek. Salah satunya adalah ketimpangan beban kerja antar anggota tim pada *sprint* awal, yang menyebabkan perlunya penyesuaian pada distribusi tugas di *sprint* berikutnya. Selain itu, keterbatasan waktu menyebabkan beberapa fitur yang direncanakan pada *sprint* keempat dan kelima belum dapat diselesaikan hingga proyek dilaporkan. Tantangan lain muncul dari kurangnya pengalaman tim dalam desain antarmuka dan kebutuhan akan perencanaan lebih matang untuk estimasi waktu pengerjaan.

5. Kesimpulan

Pada bab ini, akan disajikan kesimpulan dan saran yang dihasilkan dari penelitian dan pengembangan proyek Notares *Website* Manajemen Keuangan Cerdas dengan Metode *Scrum Agile*. Kesimpulan ini mencakup evaluasi terhadap pencapaian tujuan proyek, serta saran untuk pengembangan lebih lanjut dan penerapan metode *Scrum* dalam proyek serupa di masa depan.

5.1 Simpulan

Proyek pengembangan Notares *Website* Manajemen Keuangan Cerdas dengan Metode *Scrum Agile* berhasil mencapai beberapa tujuan utama. *Website* Notares membantu mahasiswa dalam mengelola keuangan mereka dengan fitur-fitur yang mudah digunakan, seperti pelacakan pengeluaran, perencanaan anggaran, dan rekomendasi investasi. Penggunaan metode *Scrum* memungkinkan pengembangan fitur secara iteratif dan inkremental, sehingga aplikasi dapat disesuaikan dengan kebutuhan pengguna yang dinamis. Implementasi *Scrum* juga memfasilitasi kolaborasi yang lebih baik antar anggota tim melalui pertemuan rutin seperti *Daily Scrum* dan *Sprint Review*, serta meningkatkan transparansi dalam proses pengembangan dengan adanya *Sprint Backlog* dan *Product Backlog* yang selalu diperbarui. Berdasarkan umpan balik dari pengguna, *website* Notares memberikan pengalaman pengguna yang positif dengan antarmuka yang ramah dan fitur-fitur yang relevan dengan kebutuhan manajemen keuangan mahasiswa. Selain itu, penggunaan *Next.js* sebagai *framework front-end* dan *Supabase* sebagai manajemen database mendukung pengembangan aplikasi yang responsif dan cepat, sementara algoritma rekomendasi yang digunakan dalam aplikasi membantu pengguna membuat keputusan keuangan yang lebih baik.

Metode *Scrum* memberikan kerangka kerja yang jelas dan terstruktur dalam pengembangan proyek Notares. Setiap iterasi atau *sprint* dalam *Scrum* memiliki tujuan yang spesifik, yang memastikan bahwa tim fokus pada penyelesaian fitur-fitur yang paling penting terlebih dahulu. Dengan adanya pertemuan perencanaan *sprint*, tim dapat mengidentifikasi dan memprioritaskan tugas-tugas yang perlu diselesaikan. *Daily Scrum* membantu menjaga komunikasi yang efektif di antara anggota tim, memastikan bahwa setiap anggota mengetahui tugas dan tanggung jawab masing-masing, serta membantu dalam mengatasi hambatan yang mungkin muncul selama pengembangan. Evaluasi *sprint* yang dilakukan pada akhir setiap iterasi memungkinkan tim untuk merefleksikan pencapaian dan belajar dari pengalaman, sehingga perbaikan dapat diterapkan pada *sprint* berikutnya. Penggunaan *Scrum* juga memastikan bahwa setiap perubahan atau permintaan baru dari pengguna dapat diakomodasi dengan cepat dan efisien, tanpa mengganggu alur kerja yang sudah ada.

Kebaruan dari penelitian ini terletak pada penerapan metode Agile dengan kerangka kerja *Scrum* secara penuh dalam pengembangan sistem manajemen keuangan yang ditujukan khusus untuk mahasiswa—segmen pengguna yang selama ini kurang mendapat perhatian dalam aplikasi sejenis. Relevansi aplikasi Notares juga tinggi karena mampu menjawab permasalahan nyata mahasiswa dalam mengelola keuangan pribadi secara praktis dan edukatif, melalui fitur-fitur yang disesuaikan dengan pola pengeluaran dan keterbatasan anggaran mereka.

5.2 Saran

Potensi pengembangan ke depan sangat terbuka, seperti integrasi dengan aplikasi perbankan, dompet digital, atau sistem pelaporan keuangan kampus. Selain itu, algoritma rekomendasi keuangan dapat dikembangkan lebih lanjut dengan pendekatan *machine learning* agar dapat memberikan saran keuangan yang lebih presisi. Tidak hanya terbatas pada mahasiswa, Notares juga memiliki potensi untuk diadaptasi bagi segmen lain, seperti pelajar, pekerja baru, atau komunitas dengan kebutuhan edukasi keuangan digital yang tinggi. Oleh karena itu, aplikasi ini memiliki nilai praktis yang dapat diterapkan lebih luas sebagai solusi keuangan berbasis teknologi yang inklusif dan berkelanjutan.

6. Daftar Rujukan

- [1] Muhammad Rheza Tawakkal, Girang Permata Gusti, Rudi Triadi Yulianto, Muhammad Zaini, and Udin Rinaldi, “Permasalahan Dan Solusi Perilaku Keuangan Mahasiswa Di Indonesia: Studi Deskriptif,” *J. Ekon. STIEP*, vol. 8, no. 1, pp. 140–146, 2023, doi: 10.54526/jes.v8i1.145.
- [2] D. E. Natalia, S. Murni, and V. N. Untu, “Analisis Tingkat Literasi Dan Pengelolaan Keuangan Pribadi Mahasiswa Di Fakultas Ekonomi Dan Bisnis Univeristas Sam Ratulangi,” *J. EMBA J. Ris. Ekon. Manajemen, Bisnis dan Akunt.*, vol. 7, no. 2, pp. 2131–2140, 2019, [Online]. Available: <https://doi.org/10.35794/emba.v7i2.24018>.
- [3] S. Al-Saqqa, S. Sawalha, and H. Abdelnabi, “Agile software development: Methodologies and trends,” *Int. J. Interact. Mob. Technol.*, vol. 14, no. 11, pp. 246–270, 2020, doi: 10.3991/ijim.v14i11.13269.
- [4] E. P. Wonohardjo, R. F. Sunaryo, Y. Sudiyono, N. Surantha, and Suharjo, “A systematic review of scrum in software development,” *Int. J. Informatics Vis.*, vol. 3, no. 2, pp. 108–112, 2019, doi: 10.30630/joiv.3.2.167.
- [5] J. Noordman, J. A. Driesenaar, I. R. van Bruinessen, and S. van Dulmen, “ListeningTime; participatory development of a web-based preparatory communication tool for elderly cancer patients and their healthcare providers,” *Internet Interv.*, vol. 9, no. May, pp. 51–56, 2017, doi: 10.1016/j.invent.2017.05.002.
- [6] I. Puspitasari, N. Nuzulita, and C.-S. Hsiao, “Agile User-Centered Design Framework to Support the Development of E-Health for Patient Education,” in *Computer and Information Science and Engineering: Volume 16*, Springer, 2024, pp. 131–144.
- [7] G. W. Sasmito and L. O. Mohamad Zulfiqar, “Implementation of Scrum Framework on Web Development of Mapping Salted Egg Production,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 879, no. 1, 2020, doi: 10.1088/1757-899X/879/1/012086.
- [8] M. Laila, A. Wibawa, I. Zaeni, and Z. Zuhriyah, “Implementasi SCRUM pada Pengembangan Aplikasi Sistem ADI STETSA SMAN 4 Malang: Bidang Bimbingan Konseling,” *J. Inov. Teknol. dan Edukasi Tek.*, vol. 1, no. 7, pp. 526–535, 2021, doi: 10.17977/um068v1i72021p526-535.
- [9] D. Akmaluddin, M. H. Purwanto, R. Ramdhani, and M. A. Dewi, “Agile Scrum Model for Development of Preadmission Information System Web Based at Sint Carolus Hospital Jakarta,” *Proceeding - IEEE 10th Inf. Technol. Int. Semin. ITIS 2024*, pp. 240–245, 2024, doi: 10.1109/ITIS64716.2024.10845496.
- [10] K. S. Alim, N. A. Ekowati, R. Y. Kisworini, and L. Riyandari, “Design and Development of Web-based Application Cangkingan using Scrum Method,” *J. Tek. Inform.*, vol. 4, no. 4, pp. 953–965, 2023.
- [11] E. W. Zmudzynska and H.-C. Chen, “Using scrum to build tourism information mobile application,” in *International Conference on Advanced Information Networking and Applications*, 2024, pp. 426–437.
- [12] B. Ayu, S. Muntai, E. Darwiyanto, and R. Nurtantyana, “Development of a Mobile Application Using Scrum Pattern for Monitoring Project Information System,” *Indones. J. Comput.*, vol. 9, no. August, pp. 134–153, 2024, doi: 10.34818/indoic.2024.9.2.970.
- [13] M. D. Kadenic, K. Koumaditis, and L. Junker-Jensen, “Mastering scrum with a focus on team maturity and key components of scrum,” *Inf. Softw. Technol.*, vol. 153, no. April 2022, p. 107079, 2023, doi: 10.1016/j.infsof.2022.107079.
- [14] A. Baehaqi, M. S. Basit, R. E. Indrajit, and R. D. Kurniawan, “Front End Learning Management System Development Using The Nextjs Framework,” *J. Tek. Inform.*, vol. 4, no. 4, pp. 899–911, 2023.
- [15] V. Patel, “Analyzing the Impact of NextJS on Site Performance and SEO,” *Int. J. Comput. Appl. Technol. Res.*, no. October, 2023, doi: 10.7753/ijcatr1210.1004.
- [16] S. Pati and Y. Zaki, *Evaluating the Efficacy of Next.js: A Comparative Analysis with React.js on Performance, SEO, and Global Network Equity*, vol. 1, no. 1. Association for Computing Machinery, 2025.

