JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

Smart Reminder: Sistem Pengingat Tugas Berbasis *ReactJS* dan *Google Apps Script* untuk Manajemen Proyek Tim

Akas Bagus Setiawan[1*], Mas'ud Hermansyah[2]

[1]Teknik Informatika PSDKU Sidoarjo, Jurusan Teknologi Informasi, Politeknik Negeri Jember, Indonesia
[2]Bisnis Digital, Jurusan Bisnis, Politeknik Negeri Jember, Indonesia
Email: [1]akasbagus_s@polije.ac.id, [2]mas_udhermansyah@polije.ac.id
Email Penulis Korespondensi: [1]akasbagus_s@polije.ac.id

Abstrak− Manajemen tugas yang efisien merupakan tantangan utama dalam organisasi modern, terutama dalam mengelola tenggat waktu serta memastikan kolaborasi yang optimal antaranggota tim. Penelitian ini bertujuan mengembangkan sistem otomatisasi penjadwalan dan pengingat tugas berbasis cloud dengan mengintegrasikan ReactJS sebagai frontend untuk input data, serta Google Apps Script dan Google Calendar sebagai backend. Melalui sistem ini, data tugas yang dimasukkan melalui formulir ReactJS dikirim secara real-time ke endpoint Google Apps Script, kemudian dijadwalkan secara otomatis ke Google Calendar lengkap dengan notifikasi pengingat sebelum batas waktu penyelesaian. Metodologi penelitian mencakup perancangan antarmuka pengguna berbasis prinsip UI/UX, pengembangan skrip pemrosesan otomatis di Apps Script, serta pengujian sistem dalam lingkungan kerja nyata. Hasil pengujian menunjukkan adanya peningkatan signifikan dalam kepatuhan terhadap tenggat waktu dan efisiensi pemantauan tugas, disertai penurunan beban administratif dan potensi human error pada proses penjadwalan manual. Berdasarkan hasil survei kepuasan pengguna, 87% responden menyatakan sistem ini mempermudah pengelolaan tugas dan membantu dalam mengingat deadline. Sebagian besar responden juga menilai bahwa formulir input berbasis ReactJS mudah digunakan dan validasi datanya berjalan efektif dalam meminimalkan kesalahan input. Dengan hasil tersebut, sistem berbasis ReactJS dan Apps Script ini dapat dijadikan solusi inovatif dan skalabel untuk mendukung manajemen tugas dan kolaborasi tim berbasis cloud di berbagai lingkungan organisasi.

Kata Kunci: Apps Script, Google Calendar, ReactJS, Manajemen Proyek, Otomatisasi Penjadwalan

Abstract− Efficient task management is a major challenge in modern organizations, especially in managing deadlines and ensuring optimal collaboration between team members. This research aims to develop a cloud-based task scheduling and reminder automation system by integrating ReactJS as the frontend for data input, and Google Apps Script and Google Calendar as the backend. Through this system, task data entered through the ReactJS form is sent in real-time to the Google Apps Script endpoint, then automatically scheduled to Google Calendar complete with reminder notifications before the completion deadline. The research methodology included designing a user interface based on UI/UX principles, developing an automated processing script in Apps Script, and testing the system in a real-world work environment. Test results demonstrated significant improvements in deadline adherence and task monitoring efficiency, along with a reduction in administrative burden and the potential for human error in the manual scheduling process. Based on a user satisfaction survey, 87% of respondents stated that the system simplifies task management and helps them remember deadlines. Most respondents also found the ReactJS-based input form easy to use and its data validation effective in minimizing input errors. These results suggest that this ReactJS-Apps Script-based system could be an innovative and scalable solution to support cloud-based task management and team collaboration across various organizational environments.

Keywords: Apps Script, Google Calendar, ReactJS, Project Management, Scheduling Automation

1. PENDAHULUAN

Di era digital saat ini, kerja kolaboratif menjadi elemen penting dalam penyelesaian proyek di berbagai bidang, baik di dunia industri, organisasi, maupun lembaga pendidikan. Keberhasilan sebuah proyek sangat dipengaruhi oleh efektivitas koordinasi dan komunikasi antartim [1]. Namun, dalam praktiknya, sering terjadi keterlambatan penyelesaian tugas akibat kelalaian individu, kurangnya pengingat tenggat waktu, serta pembagian tugas yang tidak teratur. Kondisi tersebut menimbulkan kebutuhan akan sistem pengingat tugas *(task reminder system)* yang dapat membantu tim mengelola pekerjaan secara efisien dan tepat waktu [2]. Perkembangan teknologi informasi telah menghadirkan berbagai sistem manajemen proyek berbasis web yang mendukung pengaturan jadwal, pembagian tugas, serta pemantauan kemajuan proyek. Penelitian yang dilakukan oleh [3], [4] terkait dengan "Rancang Bangun Sistem Manajemen Ruang Lingkup dan Jadwal Proyek Berbasis Web" menunjukkan bahwa sistem berbasis web mampu meningkatkan keteraturan jadwal kerja dan visibilitas proyek. Sistem seperti ini memberikan akses lintas platform, memudahkan kolaborasi tim, dan memastikan setiap anggota memahami tanggung jawabnya [5]. Meskipun demikian, sebagian besar penelitian terdahulu masih berfokus pada penjadwalan dan pengelolaan proyek secara umum, belum mengarah pada mekanisme pengingat otomatis yang dapat mengingatkan setiap anggota terhadap tugas spesifik yang harus diselesaikan.

Sementara itu, munculnya teknologi *Google Apps Script* (GAS) memberikan peluang untuk menciptakan sistem otomatisasi berbasis *cloud* yang efisien. GAS memungkinkan integrasi dengan berbagai layanan *Google* seperti *Sheets*, *Calendar*, dan *Gmail*. Penelitian oleh [6] berjudul "Pengembangan Sistem Informasi Pendanaan Sumber Daya Manusia

This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024
Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

Menggunakan *Google Apps Script*" menunjukkan bahwa GAS dapat meningkatkan efisiensi dan transparansi dalam pengelolaan pendanaan SDM di berbagai organisasi. Selain itu, penelitian lain oleh [7] yang mengintegrasikan *Google App Script* dengan *Codeigniter* dalam merancang aplikasi web membuktikan kemampuannya dalam menciptakan sistem informasi ringan tanpa memerlukan server eksternal. Studi lain bahkan memanfaatkan integrasi GAS dan *Google Calendar* oleh [8] untuk mengelola kegiatan organisasi dan mengirim pengingat jadwal secara otomatis kepada anggota. Meskipun teknologi tersebut menawarkan berbagai keunggulan, sebagian besar penelitian masih berfokus pada otomatisasi administrasi, penjadwalan kegiatan, atau pengingat pribadi. Penelitian seperti *Task Reminder* untuk mahasiswa misalnya, hanya menangani pengingat individu, bukan pengelolaan tugas kolaboratif dalam proyek tim [9]. Begitu pula sistem yang menggunakan *Google Calendar* umumnya digunakan untuk aktivitas rutin, bukan manajemen proyek yang memiliki struktur kerja kompleks dan keterkaitan antaranggota. Dengan demikian, masih terdapat kesenjangan dalam pengembangan sistem pengingat tugas yang bersifat kolaboratif dan terintegrasi dengan teknologi modern.

Berdasarkan analisis kesenjangan *(gap analysis)*, terdapat beberapa hal yang belum terjawab oleh penelitian terdahulu. Pertama, belum banyak sistem pengingat tugas otomatis yang dirancang khusus untuk konteks kerja tim, di mana setiap anggota menerima pengingat sesuai tanggung jawab dan tenggat waktunya. Kedua, mayoritas sistem manajemen proyek hanya menyoroti perencanaan makro, tanpa fitur pengingat *real-time* yang dapat diakses secara praktis oleh setiap anggota. Ketiga, meskipun *Google Apps Script* telah digunakan untuk otomatisasi sederhana, belum banyak penelitian yang mengombinasikannya dengan antarmuka modern seperti *ReactJS*, yang menawarkan tampilan dinamis dan responsif untuk memudahkan kolaborasi tim. Kombinasi antara *ReactJS* dan *Google Apps Script* memiliki potensi besar dalam menciptakan sistem pengingat tugas yang efisien dan ringan. *ReactJS* memberikan kemampuan antarmuka yang interaktif dan cepat [10], sedangkan *Google Apps Script* berfungsi sebagai penggerak otomatisasi di sisi *backend* yang terintegrasi dengan layanan *Google* [11]. Dengan menggabungkan kedua teknologi ini, sistem pengingat tugas dapat dikembangkan tanpa infrastruktur server yang rumit, sekaligus memberikan pengalaman pengguna yang optimal bagi anggota tim proyek.

Berdasarkan latar belakang tersebut, penelitian ini berjudul "*Smart Reminder*: Sistem Pengingat Tugas Berbasis *ReactJS dan Google Apps Script* untuk Manajemen Proyek Tim". Sistem ini dirancang untuk membantu tim proyek dalam mengelola tugas secara kolaboratif melalui fitur pengingat otomatis yang dikirimkan sesuai tenggat waktu [12]. Dengan adanya pengingat berbasis *cloud* yang terintegrasi dengan *email* atau kalender, setiap anggota dapat menerima notifikasi secara tepat waktu dan memantau progres proyek melalui antarmuka interaktif. Penelitian ini bertujuan untuk merancang dan mengembangkan sistem *Smart Reminder* berbasis *ReactJS* dan *Google Apps Script* yang dapat meningkatkan efektivitas pengelolaan tugas dalam tim proyek. Selain itu, penelitian ini juga bertujuan untuk menilai sejauh mana sistem tersebut mampu mengurangi keterlambatan pekerjaan, memperkuat koordinasi antaranggota, dan meningkatkan produktivitas tim. Dengan demikian, hasil penelitian diharapkan dapat memberikan kontribusi nyata dalam pengembangan sistem manajemen proyek modern yang cerdas, kolaboratif, dan efisien.

2. METODOLOGI PENELITIAN

Penelitian ini menggunakan pendekatan *Research and Development* (R&D), dengan fokus utama pada pengembangan aplikasi *ReactJS* sebagai antarmuka penginputan tugas, integrasi *endpoint Google Apps Script* sebagai *backend*, serta otomasi penjadwalan dengan *Google Calendar*. Langkah pengembangan sebagai berikut [13], [14]**:**

2.1 Analisis Kebutuhan

Analisis kebutuhan dilakukan melalui survei terhadap pengguna di lingkungan Jurusan Teknologi Informasi, Politeknik Negeri Jember, untuk mengidentifikasi keterbatasan sistem manual serta kebutuhan akan digitalisasi proses input tugas, terutama pada kegiatan Pengabdian kepada Masyarakat. Data esensial yang dikumpulkan meliputi nama PIC, *email*, tugas, deskripsi, prioritas, dan *deadline*. Analisis ini menjadi dasar desain aplikasi *ReactJS* dengan *field* dan validasi yang sesuai kebutuhan pengguna tim proyek [15].

2.2 Perancangan Sistem

Sistem dirancang secara modular, dengan pengembangan aplikasi *ReactJS* sebagai *frontend* yang fokus pada kemudahan *input*, validasi data *real-time*, dan desain *UI/UX modern*. Komponen *form ReactJS* mengadopsi *state* dan *controlled input*, serta *error handling* sesuai *best practice frontend development*.

2.3 Pengembangan Aplikasi

Aplikasi *ReactJS* dikembangkan menggunakan pendekatan *agile*, menerapkan validasi di sisi *frontend* (*required*, *email* format) dan *feedback* status pengiriman. Setelah data diisikan dan divalidasi, aplikasi melakukan HTTP POST ke *endpoint Apps Script* berbasis API, yang memproses dan meneruskan data ke *Google Sheets* dan *Calendar*. Pengujian dilakukan untuk memastikan data yang dikirim ke *endpoint* sesuai dengan skema *backend* beserta response status [16].

2.4 Integrasi *Backend*

This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024 Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

Endpoint Google Apps Script berfungsi sebagai penerima *request* dari *ReactJS*. *Script* mengolah data *input*, menyimpannya ke *Google Sheets*, serta membuat *event calendar* beserta notifikasi pengingat *deadline* secara otomatis sesuai logika bisnis terdefinisi.

2.5 Implementasi dan Pengujian

Implementasi dilakukan pada lingkungan kerja nyata yang melibatkan pengguna sesuai *role*, diuji baik di *desktop* maupun perangkat *mobile*. Pengujian *usability* dilakukan untuk menilai kemudahan penggunaan, validasi *input*, dan efektivitas *feedback* kepada pengguna. *Error handling* baik pada level *frontend (ReactJS)* maupun *backend (Apps Script)* dikaji untuk memastikan keandalan sistem [17].

2.6 Evaluasi Sistem

Proses untuk menilai keefektifan dan akurasi dari identifikasi masalah, kendala, dan risiko yang terjadi selama siklus hidup pengembangan sistem. Proses ini bertujuan untuk memastikan bahwa semua faktor penghambat yang berpotensi menghambat keberhasilan proyek telah teridentifikasi, dianalisis, dan diprioritaskan dengan benar, sehingga dapat ditangani secara proaktif.

3. HASIL DAN PEMBAHASAN

Bagian ini membahas hasil analisis kebutuhan, perancangan, pengembangan, serta implementasi sistem *Smart Reminder* yang dikembangkan untuk mendukung manajemen proyek tim.

3.1 Analisis Kebutuhan

Mengidentifikasi kebutuhan pengguna menjadi langkah awal yang penting dalam proses pengembangan sistem otomatisasi penjadwalan dan pengingat tugas. Melalui survei internal yang dilakukan terhadap 15 partisipan di lingkungan Jurusan Teknologi Informasi, Politeknik Negeri Jember, ditemukan bahwa mayoritas anggota tim mengalami kendala dalam mengelola jadwal tugas secara efisien. Sebanyak 60% responden menyatakan bahwa mereka sering lupa tenggat waktu tugas karena kurangnya sistem pengingat yang terintegrasi. Hasil ini konsisten dengan penelitian terdahulu yang menunjukkan bahwa 65% profesional mengalami kesulitan dalam mengelola deadline tanpa sistem otomatisasi [18].

Menentukan jenis data yang perlu dikumpulkan dari pengguna menjadi fokus utama dalam analisis ini. Hasil survei menunjukkan bahwa data esensial yang diperlukan terlihat pada Tabel 1. Pendekatan ini mengikuti best practice dalam pengembangan sistem informasi, di mana identifikasi kebutuhan data menjadi langkah kritis dalam perancangan sistem [19].

Tabel 1. Kebutuhan Data Form pada Aplikasi

Data	Tipe Data
Nama PIC	*Varchar*
Email PIC	*Varchar*
Tugas PIC	*Varchar*
Deskripsi	*Varchar*
Prioritas	*Varchar*
Deadline	*Datetime*

3.2 Perancangan Sistem

Sistem terdiri dari tiga komponen utama: Form aplikasi ReactJS sebagai media input, Google Sheet sebagai basis data sementara, dan Google Calendar sebagai platform penjadwalan tugas. Proses utama yang dirancang meliputi pengambilan data, pemrosesan, dan pengiriman event ke kalender. Langkah-langkah proses tersebut dijelaskan melalui pseudocode pada Tabel 2. Arsitektur sistem ini mengikuti prinsip modular design yang memungkinkan pengembangan dan maintenance yang lebih mudah [20].

Tabel 2. Pseudocode (handleFormSubmit(e))

No.	*Pseudocode*
1	Ambil data dari Form ReactJS yang dikirim ke backend (Apps Script)
2	Tentukan baris tempat data baru disubmit
3	Ambil data dari kolom berikut di baris tersebut:
	- Nama ← Kolom ke-2
	- Email ← Kolom ke-3
	- Tugas ← Kolom ke-4

This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024
Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

	- Deskripsi ← Kolom ke-5 - Prioritas ← Kolom ke-6 - Deadline ← Kolom ke-7
4	Akses kalender default milik pengguna
5	Ubah deadline (string) menjadi objek tanggal
6	Buat event baru di kalender: - Judul: "📌 Deadline: [Tugas] - [Nama]" - Waktu mulai: jam 08.00 di tanggal deadline - Waktu selesai: jam 09.00 di tanggal deadline - Keterangan (description): isi dengan Tugas, Deskripsi, dan Prioritas - Tambahkan tamu (guest): Email - Kirim undangan secara otomatis
7	Tambahkan notifikasi popup 1 hari (1440 menit) sebelum event

Implementasi pseudocode yang disajikan pada Tabel 2 mendemonstrasikan proses ekstraksi data terstruktur dari kolom kedua hingga ketujuh. Data yang diekstraksi secara spesifik meliputi nama PIC, alamat Email, jenis tugas, deskripsi, tingkat prioritas, dan tanggal jatuh tempo *(deadline)*. Setelah berhasil diambil, data ini selanjutnya diolah untuk menginisiasi pembuatan acara baru pada layanan *Google Calendar* milik pengguna. Struktur judul acara tersebut diformat secara konsisten, mengintegrasikan sebuah ikon, nama tugas, dan nama PIC. Penjadwalan acara telah ditetapkan berlangsung dari pukul 08.00 hingga 09.00 pada tanggal tenggat, dengan bagian deskripsi acara yang memuat rincian tentang tugas, deskripsi, dan prioritas yang bersangkutan. Alamat Email PIC secara otomatis ditambahkan sebagai tamu acara, memfasilitasi pengiriman undangan secara otomatis. Sebagai langkah tambahan, sistem juga mengintegrasikan notifikasi *popup* sebagai pengingat tugas yang dikirimkan satu hari (1440 menit) sebelum jadwal yang ditentukan. Pendekatan operasional ini selaras dengan prinsip arsitektur berbasis peristiwa *(event-driven architecture)*, yang mendukung kapabilitas respons *real-time* terhadap modifikasi data [21].

Selanjutnya, dibuatkan proses untuk mengatur pemicu otomatis (trigger) pada spreadsheet yang terhubung dengan Form aplikasi ReactJS (bukan lagi Google Form). Langkah-langkah proses tersebut dijelaskan melalui pseudocode pada Tabel 3.

Tabel 3. Pseudocode(createTrigger())

No.	*Pseudocode*
1	Ambil spreadsheet aktif
2	Buat trigger baru: - Nama fungsi yang dipanggil: "handleFormSubmit" - Tipe pemicu: saat form disubmit - Target: spreadsheet aktif
3	// Fungsi utama yang dijalankan saat form disubmit function handleFormSubmit(e) { const sheet = e.source.getActiveSheet(); const row = e.range.getRow(); const nama = sheet.getRange(row, 2).getValue(); const email = sheet.getRange(row, 3).getValue(); const tugas = sheet.getRange(row, 4).getValue(); const deskripsi = sheet.getRange(row, 5).getValue(); const prioritas = sheet.getRange(row, 6).getValue(); const deadline = sheet.getRange(row, 7).getValue(); const calendar = CalendarApp.getDefaultCalendar(); const deadlineDate = new Date(deadline); const event = calendar.createEvent(`📌 Deadline: ${tugas} - ${nama}`, new Date(deadlineDate.getFullYear(), deadlineDate.getMonth(), deadlineDate.getDate(), 8, 0), new Date(deadlineDate.getFullYear(), deadlineDate.getMonth(), deadlineDate.getDate(), 9, 0), { description: `Tugas: ${tugas}\nDeskripsi: ${deskripsi}\nPrioritas: ${prioritas}`, guests: email, sendInvites: true });

This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024 Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

```
  event.addPopupReminder(1440);
}

// Fungsi sekali jalan untuk membuat trigger otomatis
function createTrigger() {
  const ss = SpreadsheetApp.getActive();
  ScriptApp.newTrigger("handleFormSubmit")
    .forSpreadsheet(ss)
    .onFormSubmit()
    .create();
}
```

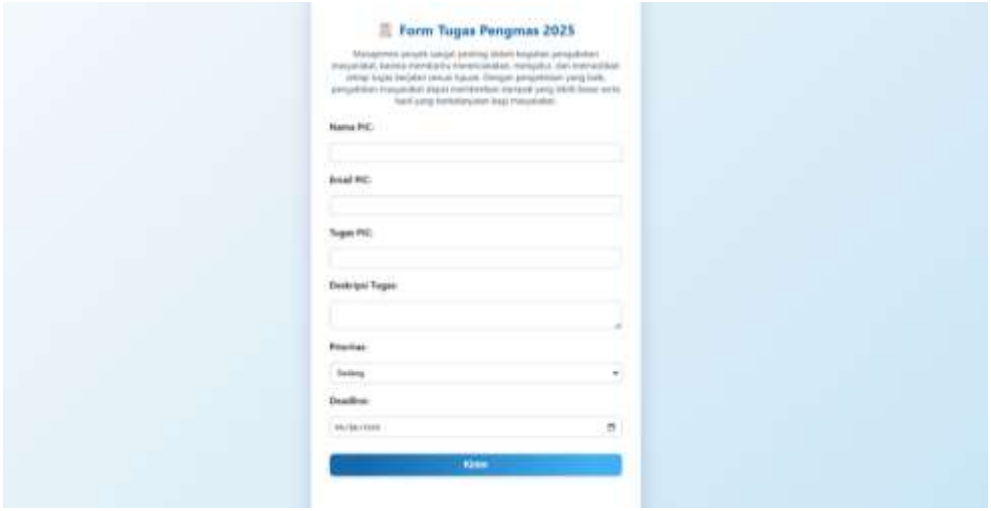
4	Simpan trigger tersebut agar aktif secara otomatis

Pseudocode pada Tabel 3 menjelaskan bahwa langkah pertama adalah mendapatkan data yang dikirim dari Form aplikasi ReactJS ke endpoint Apps Script. Selanjutnya, sistem membuat trigger baru dengan menentukan bahwa fungsi yang akan dijalankan adalah "handleFormSubmit", yaitu fungsi yang menangani logika saat formulir disubmit (khusus fallback atau maintenance lama). Tipe pemicu yang digunakan adalah "on form submit" untuk skenario legacy. Meskipun demikian, dalam arsitektur sistem akhir, disarankan agar mekanisme pengiriman (submit) data dilakukan melalui metode HTTP POST ke endpoint doPost. Penerapan ini memanfaatkan kerangka kerja frontend yang lebih mutakhir dan robust, seperti ReactJS. Strategi implementasi ini selaras dengan prinsip desain otomasi-prioritas (automation-first) dan pendekatan berbasis-peristiwa (event-driven), yang secara signifikan meminimalisasi kebutuhan akan intervensi operasional manual [22].

3.3 Pengembangan Aplikasi

Pengembangan frontend aplikasi dilakukan dengan memanfaatkan ReactJS sebagai framework utama. Tahapan awal dimulai dengan inisialisasi proyek ReactJS, misalnya melalui perintah create-react-app, diikuti dengan perancangan antarmuka formulir berbasis komponen (component-based form design). Setiap field input yang dibutuhkan seperti Nama PIC, Email, Tugas, Deskripsi, Prioritas, dan Deadline diimplementasikan menggunakan controlled components berbasis state, sehingga setiap perubahan data dapat dikelola dan dipantau secara real-time. Pendekatan ini memungkinkan sistem untuk mempertahankan konsistensi data antar-komponen serta meningkatkan efisiensi dalam pengelolaan interaksi pengguna.

Selanjutnya, proses validasi data dilakukan langsung pada sisi frontend guna memastikan kelengkapan isian, kesesuaian format email, serta keabsahan tanggal sebelum data dikirim ke backend. Pengiriman data dilakukan menggunakan metode HTTP POST melalui fetch API menuju endpoint Google Apps Script dalam format JSON sesuai kebutuhan backend. Setelah proses pengiriman selesai, sistem memberikan umpan balik (feedback) berupa notifikasi status pengiriman kepada pengguna, baik berhasil maupun gagal, dan secara otomatis mereset formulir apabila data berhasil disimpan. Seluruh tahapan ini diuji untuk menjamin validitas input, keandalan proses transmisi data, serta kenyamanan antarmuka pengguna. Dengan implementasi tersebut, frontend ReactJS mampu mengakomodasi input tugas, menjalankan validasi data, serta mengirimkan informasi secara efektif ke backend Apps Script yang selanjutnya terintegrasi dengan Google Calendar untuk otomasi penjadwalan dan pengingat tugas tim. Hasil dari pengembangan frontend menggunakan ReactJS terlihat pada Gambar 1.



Gambar 1. Tampilan Aplikasi menggunakan ReactJS

This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024 Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

3.4 Integrasi Backend

Mengembangkan Google Apps Script menjadi tahap inti dalam mewujudkan otomatisasi penjadwalan dan pengingat tugas. Script dirancang untuk merespons setiap kali data baru dikirimkan melalui Form aplikasi ReactJS, dengan memanfaatkan fungsi pemicu otomatis (trigger). Fungsi ini dijalankan secara real-time setelah input dari frontend ReactJS, memungkinkan sistem segera memproses input tanpa intervensi manual. Detail code untuk pengembangan Google Apps Script tersaji pada Gambar 2. Pengembangan ini mengikuti prinsip rapid prototyping yang memungkinkan pengujian fitur secara cepat [23].

```
function formatDateToIndonesian(dateStr) {
  const date = new Date(dateStr);
  if (isNaN(date)) return dateStr; // Fallback jika format tidak valid
  const day = String(date.getDate()).padStart(2, '0');
  const month = String(date.getMonth() + 1).padStart(2, '0');
  const year = date.getFullYear();
  return `${day}/${month}/${year}`;
}

//  Fungsi utama dari form Google (biarkan tetap ada)
function handleFormSubmit(e) {
  const sheet = e.source.getActiveSheet();
  const row = e.range.getRow();

  const nama = sheet.getRange(row, 2).getValue();
  const email = sheet.getRange(row, 3).getValue();
  const tugas = sheet.getRange(row, 4).getValue();
  const deskripsi = sheet.getRange(row, 5).getValue();
  const prioritas = sheet.getRange(row, 6).getValue();
  const deadline = sheet.getRange(row, 7).getValue();

  createCalendarEvent(nama, email, tugas, deskripsi, prioritas, deadline);
}

// Fungsi sekali jalan untuk membuat trigger otomatis (biarkan juga)
function createTrigger() {
  const ss = SpreadsheetApp.getActive();
  ScriptApp.newTrigger("handleFormSubmit")
    .forSpreadsheet(ss)
    .onFormSubmit()
    .create();
}

// Fungsi baru untuk membuat event di Google Calendar
function createCalendarEvent(nama, email, tugas, deskripsi, prioritas, deadline) {
  const calendar = CalendarApp.getDefaultCalendar();
  const deadlineDate = new Date(deadline);

  if (isNaN(deadlineDate)) return; // skip kalau tanggal tidak valid

  const event = calendar.createEvent(
    `Deadline: ${tugas} - ${nama}`,
    new Date(deadlineDate.getFullYear(), deadlineDate.getMonth(), deadlineDate.getDate(), 8, 0),
    new Date(deadlineDate.getFullYear(), deadlineDate.getMonth(), deadlineDate.getDate(), 9, 0),
    {
      description: `Tugas: ${tugas}\nDeskripsi: ${deskripsi}\nPrioritas: ${prioritas}`,
      guests: email,
      sendInvites: true
    }
  );

  // Reminder popup 1 hari sebelumnya
  event.addPopupReminder(1440);
}

// Fungsi API untuk menerima data dari React
function doPost(e) {
  try {
    const sheet = SpreadsheetApp.getActiveSpreadsheet().getSheetByName("Form Responses 1");
    const data = JSON.parse(e.postData.contents);

    const timestamp = new Date();
    const formattedDeadline = formatDateToIndonesian(data.deadline);

    const newRow = [
      timestamp,
      data.namaPIC,
      data.emailPIC,
      data.tugasPIC,
      data.deskripsi,
      data.prioritas,
      formattedDeadline
    ];

    // Menambahkan data baru ke sheet
    sheet.appendRow(newRow);

    // Otomatis bikin event di Google Calendar juga
    createCalendarEvent(
      data.namaPIC,
      data.emailPIC,
      data.tugasPIC,
      data.deskripsi,
      data.prioritas,
      data.deadline
    );

    return ContentService
      .createTextOutput(JSON.stringify({ result: "success", row: newRow }))
      .setMimeType(ContentService.MimeType.JSON);

  } catch (err) {
    return ContentService
      .createTextOutput(JSON.stringify({ result: "error", message: err.toString() }))
      .setMimeType(ContentService.MimeType.JSON);
  }
}
```

Gambar 2. Code Pengembangan Google Apps Script

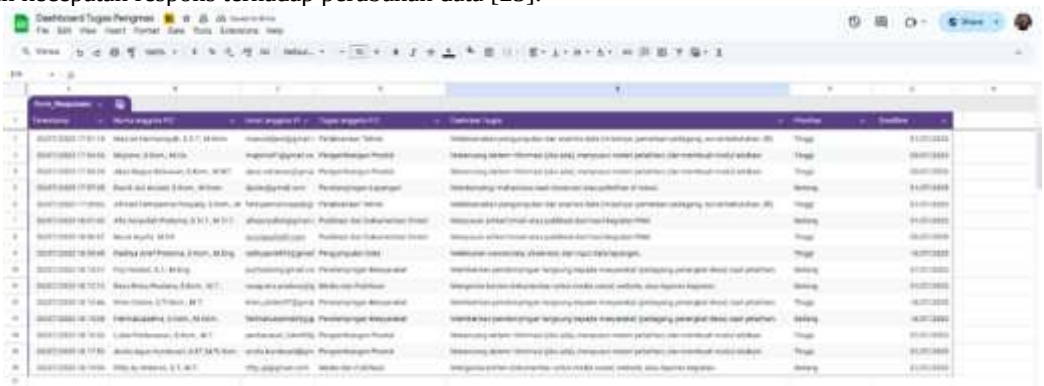
This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024 Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

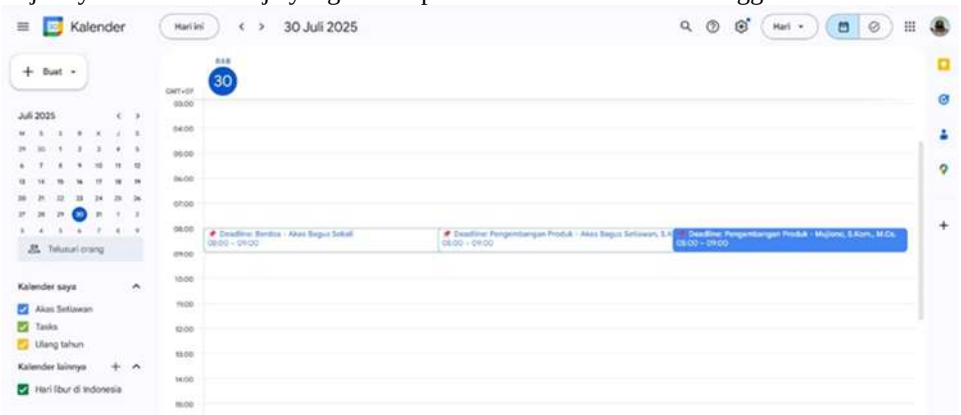
3.5 Implementasi dan Pengujian

Setiap data yang diinput mulai dari nama penanggung jawab, deskripsi tugas, hingga tanggal tenggat diverifikasi baik di sisi frontend maupun backend, menghasilkan informasi penjadwalan yang lebih akurat dan otomatis tersinkronisasi pada Google Calendar anggota tim serta manajer proyek. Temuan ini memperlihatkan bahwa tingkat penerimaan (user acceptance) terhadap sistem digital berbasis ReactJS jauh lebih tinggi dibandingkan dengan pendekatan manual maupun formulir konvensional sebelumnya [24]. Data penjadwalan yang divisualisasikan pada Gambar 3 dihasilkan melalui input pengguna pada aplikasi ReactJS, kemudian diproses secara otomatis menggunakan Google Apps Script. Data tersebut mencakup komponen penting seperti nama tugas, pihak yang bertanggung jawab, serta waktu tenggat penyelesaian. Setelah diproses, sistem secara langsung mengintegrasikan informasi tersebut ke dalam Google Calendar masing-masing anggota secara real-time, sehingga setiap individu menerima notifikasi pengingat otomatis sebelum batas waktu yang ditentukan. Pendekatan ini secara signifikan meningkatkan kesadaran tanggung jawab individu dan mengurangi potensi keterlambatan penyelesaian tugas, sejalan dengan prinsip real-time data processing yang menekankan kecepatan respons terhadap perubahan data [25].



Gambar 3. Data Penjadwalan pada *Google Spreadsheet*

Selain itu, sistem secara otomatis mengirimkan replika data penjadwalan ke akun Google Calendar milik Penanggung Jawab (PIC) proyek, sebagaimana diperlihatkan pada Gambar 4. Melalui mekanisme ini, PIC memiliki kemampuan untuk memantau seluruh aktivitas dan jadwal tugas anggota tim secara terpusat, sehingga memperoleh pandangan komprehensif terhadap distribusi pekerjaan dan alur pelaksanaan proyek yang sedang berlangsung. Implementasi integrasi tersebut berkontribusi secara signifikan terhadap peningkatan efisiensi manajemen proyek serta mendukung terwujudnya koordinasi kerja yang lebih optimal dan terstruktur antaranggota tim.



Gambar 4. Dashboard Google Calender PIC Proyek

Ketika jadwal telah tersinkronisasi dengan kalender pada perangkat seluler pengguna, sistem akan secara otomatis menampilkan pengingat berupa notifikasi yang terhubung langsung dengan layanan Google Calendar. Notifikasi tersebut tidak hanya hadir dalam bentuk tampilan visual, tetapi juga dapat dilengkapi dengan isyarat suara atau getaran, menyesuaikan dengan konfigurasi pada perangkat pengguna. Mekanisme ini memberikan peringatan tambahan yang efektif, khususnya bagi pengguna yang memiliki mobilitas tinggi atau tidak sedang mengoperasikan komputer.

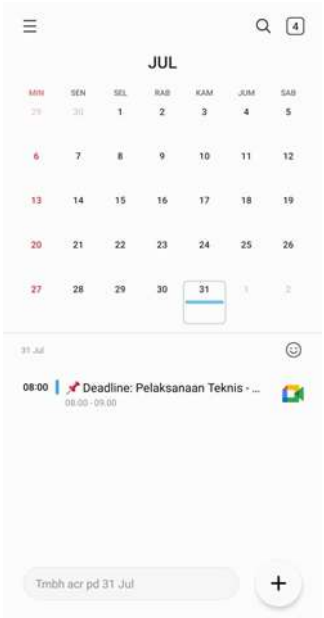
Fitur pengingat melalui perangkat seluler ini berperan penting dalam meningkatkan kepatuhan anggota tim terhadap jadwal dan tenggat waktu yang telah ditetapkan, sekaligus membantu menjaga konsistensi penyelesaian tugas secara tepat waktu. Dengan adanya sistem notifikasi lintas perangkat ini, setiap pengguna dapat tetap terinformasi tanpa bergantung pada satu kanal akses tertentu. Implementasi tersebut sejalan dengan prinsip multi-channel notification yang

This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024
Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

menekankan pentingnya ketersediaan pengingat melalui berbagai media dan perangkat untuk memastikan informasi penting tersampaikan secara efisien dan tepat waktu.



Gambar 5. Pengingat pada Kalender di Smartphone

Sebagaimana diperlihatkan pada Gambar 5, ketika jadwal telah tersinkronisasi ke kalender pada perangkat seluler pengguna, sistem akan secara otomatis menampilkan pengingat dalam bentuk notifikasi yang terhubung langsung dengan Google Calendar. Notifikasi tersebut tidak hanya berfungsi sebagai penanda visual, tetapi juga dapat disertai dengan efek suara atau getaran, bergantung pada konfigurasi perangkat masing-masing pengguna. Mekanisme ini memberikan peringatan tambahan yang efektif, memastikan bahwa pengguna tetap memperoleh informasi tepat waktu terkait tugas yang harus diselesaikan [26].

Fitur pengingat pada perangkat seluler ini terbukti berperan penting dalam membantu setiap anggota tim menjaga kewaspadaan terhadap tenggat waktu pekerjaan, khususnya bagi mereka yang memiliki tingkat mobilitas tinggi atau tidak sedang mengakses komputer. Oleh karena itu, keberadaan pengingat berbasis smartphone menjadi elemen pendukung yang krusial dalam meningkatkan kedisiplinan terhadap jadwal kerja serta memastikan penyelesaian tugas dilakukan secara tepat waktu dan terorganisasi.

3.6 Evaluasi Sistem

Hasil survei kepuasan pengguna menunjukkan bahwa sejumlah 87% responden menyatakan sistem ini mempermudah proses pengelolaan tugas sekaligus membantu dalam mengingat batas waktu penyelesaian. Sebagian besar responden menilai bahwa formulir input berbasis ReactJS memberikan kemudahan dalam penggunaan, dengan mekanisme validasi yang berfungsi secara efektif sehingga mampu meminimalkan kesalahan pada proses pengisian data. Selain itu, fitur umpan balik terhadap status pengiriman memberikan transparansi dan kejelasan terhadap proses yang sedang berlangsung, sehingga meningkatkan pengalaman pengguna secara keseluruhan [27]. Fitur pengingat berbasis popup juga memperoleh penilaian tinggi karena dinilai sangat membantu, terutama bagi anggota tim yang memiliki beban tugas dalam jumlah besar. Lebih lanjut, integrasi menyeluruh antara ReactJS, Google Apps Script, dan Google Calendar memungkinkan proses pemantauan tugas dilakukan secara real-time oleh manajer proyek [28].

Pada proses pengembangan dan implementasi sistem terintegrasi antara aplikasi ReactJS, Google Apps Script, dan Google Calendar, tim pengembang menghadapi sejumlah tantangan baik dari sisi teknis maupun non-teknis. Salah satu kendala utama yang muncul berkaitan dengan ketidaksesuaian format tanggal. Perbedaan antara format tanggal yang dihasilkan oleh frontend ReactJS dan format yang diterima oleh Google Calendar seringkali menimbulkan kesalahan pada proses penjadwalan acara. Untuk mengatasi permasalahan tersebut, diterapkan mekanisme validasi serta konversi format tanggal secara otomatis melalui skrip pemrograman pada Google Apps Script, sehingga sistem dapat menyesuaikan data sebelum dikirim ke kalender.

Selain itu, tantangan lain muncul pada aspek hak akses Google Calendar. Ditemukan bahwa beberapa anggota tim belum memberikan izin berbagi kalender, yang menyebabkan undangan kegiatan tidak terkirim atau tidak muncul pada kalender penerima. Sebagai solusi, dilakukan pendekatan sosialisasi dan pelatihan singkat kepada seluruh anggota tim untuk memberikan pemahaman terkait pengaturan izin akses dan berbagi kalender agar proses integrasi berjalan optimal. Langkah ini sejalan dengan prinsip user training yang menekankan pentingnya kesiapan dan literasi pengguna sebagai faktor kunci dalam keberhasilan implementasi sistem berbasis kolaboratif.

This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024
Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

JURNAL MEDIA INFORMATIKA [JUMIN]
Volume 6 No. 6 November – Desember 2025, Page 2934-2943
ISSN 2808-005X (media online)
Available Online at http://ejournal.sisfokomtek.org/index.php/jumin

4. KESIMPULAN

Integrasi antara ReactJS, Google Apps Script, dan Google Calendar dalam penelitian ini terbukti efektif sebagai solusi otomatisasi penjadwalan tugas dan pengingat tenggat waktu untuk mendukung manajemen proyek tim. Pemanfaatan ReactJS sebagai formulir digital meningkatkan pengalaman pengguna serta akurasi data, sedangkan otomatisasi melalui Apps Script dan notifikasi Google Calendar menghasilkan efisiensi signifikan dalam pengelolaan jadwal sekaligus mengurangi potensi kesalahan dan keterlambatan. Sistem berbasis teknologi open-source dan cloud ini layak diadopsi oleh institusi pendidikan maupun organisasi profesional berskala kecil hingga menengah karena minim biaya lisensi serta mendukung pengelolaan tugas yang modern dan scalable. Secara praktis, sistem ini memberikan dampak positif bagi peningkatan efisiensi, akuntabilitas, dan kolaborasi organisasi. Di lingkungan pendidikan, sistem ini dapat menjadi solusi cost-effective untuk manajemen proyek penelitian dan tugas akademik. Bagi lembaga pemerintahan maupun sektor bisnis seperti startup dan UMKM, sistem ini berpotensi mengoptimalkan sumber daya manusia melalui pengurangan beban administratif manual. Ke depan, pengembangan sistem dapat diarahkan pada integrasi notifikasi multi-channel (WhatsApp, Email, Telegram), dashboard progres berbasis Google Data Studio, serta penerapan Kecerdasan Buatan untuk prediksi risiko keterlambatan dan rekomendasi prioritas tugas, guna mendorong digitalisasi administrasi yang lebih adaptif dan berkelanjutan.

UCAPAN TERIMAKASIH

Penulis menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada seluruh staf dosen Jurusan Teknologi Informasi, Politeknik Negeri Jember, atas dukungan, bimbingan, serta motivasi yang telah diberikan selama proses penulisan artikel ilmiah ini. Ucapan terima kasih juga disampaikan atas berbagai masukan konstruktif dan pendampingan akademik yang membantu penulis dalam menyempurnakan konsep, metodologi, serta penyusunan hasil penelitian ini. Dukungan lingkungan akademik yang kolaboratif di Jurusan Teknologi Informasi telah memberikan inspirasi dan semangat bagi penulis untuk terus berinovasi dalam pengembangan teknologi berbasis sistem informasi. Semoga karya ilmiah ini dapat memberikan kontribusi positif bagi pengembangan ilmu pengetahuan serta menjadi bagian dari upaya peningkatan kualitas penelitian di Politeknik Negeri Jember.

REFERENCES

[1] S. Sarju, D. V. Asmarayani, and N. C. Kresnanto, “Penilaian Efektivitas Implementasi Building Information Modelling (Bim) Pada Proyek Konstruksi Bangunan Gedung,” *J. Tek. Sipil*, vol. 16, no. 4, pp. 247–260, 2022, doi: 10.24002/jts.v16i4.5539.

[2] R. Afif *et al.*, “Rancang Bangun Aplikasi Manajemen Tugas Menggunakan Laravel,” *J. Teknol. Inf. dan Komun.*, vol. 16, no. 1, pp. 6–13, 2025, doi: 10.51903/jtikp.v16i1.907.

[3] A. V. Suwandana and A. W. Utami, “Rancang Bangun Sistem Informasi Manajemen Proyek Berbasis Website Menggunakan Project Management Body Of Knowledge 6 (Studi Kasus PT. Tekno Mandala Kreatif),” *J. Emerg. Inf. Syst. Bus. Intell.*, vol. 3, no. 4, pp. 80–89, 2022, doi: 10.26740/jeisbi.v3i4.49162.

[4] F. Anzari, P. Kusumo, and R. Mulyani, “Rancang Bangun Sistem Manajemen Ruang Lingkup dan Jadwal Proyek Berbasis Web,” *MIFORTEKH (Jurnal Manaj. Inform. Teknol.*, vol. 5, no. 1, pp. 276–286, 2025.

[5] B. Kharisma and N. Santoso, “Pengembangan Aplikasi Manajemen Proyek Perangkat Lunak Kolaboratif Menggunakan Scrum,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 3, pp. 723–732, 2020.

[6] S. Wulan and T. Dirgahayu, “Pengembangan Sistem Informasi Pendanaan Sumber Daya Manusia Menggunakan Google Apps Script,” *EDUSAINTEK J. Pendidikan, Sains dan Teknol.*, vol. 11, no. 4, pp. 2238–2254, 2024, doi: 10.47668/edusaintek.v11i4.1448.

[7] T. Wijaya, “Pemanfaatan Google App Script dalam Merancang Aplikasi Web Guna Meningkatkan Efisiensi Biaya Perusahaan,” *Technol. Entrep.*, vol. 2, no. 3, pp. 8–20, 2024, [Online]. Available: https://doi.org/10.59841/ignite.v2i3.1548

[8] A. B. Setiawan and M. Hermansyah, “Otomatisasi Penjadwalan dan Pengingat Tugas Berbasis Google Form, Apps Script, dan Google Calendar untuk Efisiensi Manajemen Proyek,” *J. Sist. Inf. dan Sains Teknol. (SISTEK TRILOGI)*, vol. 7, no. 2, pp. 162–178, 2025.

[9] A. D. Irawan and W. S. Utami, “Aplikasi Reminder Jadwal Kuliah dan Tugas Mahasiswa Berbasis Android,” *J. Komput. dan Inform.*, vol. 5, no. 2, pp. 288–300, 2023.

[10] M. B. Pramadipta, M. N. Saiholau, W. E. Sulistiono, and D. Budiyanto, “Rancang Bangun Frontend Website Untuk Pemungutan Suara dengan Menggunakan ReactJs,” *JITET J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 2, pp. 1131–1140, 2022.

[11] L. R. Rosanti and G. Swalaganata, “Implementasi Google App Script untuk Input Data pada Database Master Data,” *Ris. dan E-Jurnal Manaj. Inform. Komput.*, vol. 8, no. 1, pp. 117–129, 2024, [Online]. Available: http://doi.org/10.33395/remik.v8i1.13273

[12] S. Padang, A. J. Sidauru, E. Tambunan, M. Y. Manullang, and J. Simanjuntak, “Perancangan Aplikasi Manajemen Waktu untuk Meningkatkan Produktivitas Pengguna,” *Remik Ris. dan E-Jurnal Manaj. Inform.*

This is an open access article under the CC–BY-SA license

Terakreditasi SINTA 5 SK :72/E/KPT/2024
Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**

Komput., vol. 9, no. 2, pp. 488–499, 2025.
[13] H. Halimatussa'diah, J. Julizal, and A. Irawan, "Metode Research and Development (R&D) Pelayanan Pengantar Masyarakat untuk Masyarakat Menggunakan Java," *Remik Ris. dan E-jurnal Manaj. Inform. Komput.*, vol. 8, no. April, pp. 1–16, 2024.
[14] S. Fikri *et al.*, "Perancangan Sistem Penjualan Sayuran Menggunakan React JS," *JRKT J. Rekayasa Komputasi Terap.*, vol. 5, no. 01, pp. 60–65, 2025, doi: 10.30998/jrkt.v5i01.13668.
[15] M. Abas and N. Santoso, "Pengembangan Aplikasi Manajemen Pengawasan Proyek Konstruksi berbasis Android pada Perusahaan AMF-Haq Engineering and Consultant," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 9, pp. 3799–3806, 2021, [Online]. Available: http://j-ptiik.ub.ac.id
[16] M. R. Athila, R. B. V. Sibuea, and G. A. Hutagalung, "Perancangan & Pembuatan Backend Sistem Antrian Pasien Pada Praktik Dokter Irsan Lubis Menggunakan Express Js," in *Konferensi Nasional Social dan Engineering Politeknik Negeri Medan*, 2024, pp. 499–519.
[17] A. Sah, S. D. Saraswati, N. Heriyani, and M. Badaruddin, "Pengembangan Sistem Pengeloaan Bimbingan Belajar Menggunakan Pendekatan Rapid Application Development," *J. Comput. Informatics Res.*, vol. 4, no. 1, pp. 271–283, 2024, doi: 10.47065/comforch.v4i1.1409.
[18] O. Hamdito, I. Indarto, and Y. Budiati, "Pengaruh Kerjasama Tim, Kompetensi dan Birokrasi Terhadap Kinerja Anggota Satlantas Polrestabes Semarang dengan Mediasi Kepuasan Kerja," *J. Litbang Polri*, vol. 28, no. 1, pp. 35–55, 2025, doi: 10.46976/litbangpolri.v28i1.296.
[19] F. Ibrahim, T. R. Agus, and N. W. W. Sari, "Identifikasi Metode Pengembangan Sistem Informasi di Indonesia: A Systematic Literature Review," *Metik J.*, vol. 5, no. 1, pp. 47–54, 2021, doi: 10.47002/metik.v5i1.215.
[20] M. A. Zihad and I. Nuryasin, "Implementasi Clean Architecture pada Aplikasi Mobile Al-Quran Berbasis Flutter," *JOISIE J. Inf. Syst. Informatics Eng.*, vol. 8, no. 2, pp. 369–380, 2024, [Online]. Available: https://doi.org/10.35145/joisie.v8i2.4763
[21] U. Syarif and P. Pizaini, "Penerapan Event-Driven Microservices Pada Aplikasi Layanan Penerimaan Peserta Didik Baru," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 07, no. September, pp. 745–756, 2022.
[22] S. N. Salim, A. Wirawan, and I. P. Wardhani, "Evaluasi Rancangan Antarmuka HCI Modern Berbasis Kecerdasan Buatan," *J. Ilm. KOMPUTASI*, vol. 23, no. 4, pp. 531–538, 2024.
[23] M. Syarif and D. Risdiansyah, "Pemanfaatan Metode Prototype Dalam Perancangan Sistem Informasi Penjualan Berbasis Website," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 4, pp. 7945–7952, 2024, doi: 10.36040/jati.v8i4.10467.
[24] M. Nuraminudin, M. M. Dewi, S. Suharsono, A. Dahlan, and L. Lukman, "Implementasi Reactjs Pada Pembuatan Sistem Informasi Digital Printing Berbasis Website," *Inf. Syst. J.*, vol. 6, no. 01, pp. 25–32, 2023, doi: 10.24076/infosjournal.2023v6i01.1214.
[25] A. Faroqi and T. L. M. Suryanto, "Pemanfaatan Google Calendar untuk Pembuatan Kalender Akademik di SMP Miftahul Ulum Surabaya," *J. Layanan Masy. (Journal Public Serv.*, vol. 4, no. 1, pp. 13–16, 2020.
[26] A. Wahyudin, A. Anisyah, and D. Ahmaddifa, "Pengembangan Sistem Notifikasi Real Time untuk Aplikasi Manajemen Persuratan Multiplatform menggunakan Firebase Cloud Messaging dan Application Programming Interface," *J. Tek. Inform. Unika ST. Thomas*, vol. 09, no. 02, pp. 2657–1501, 2024.
[27] S. Rahmadhani, D. W. Wildana, H. W. Arumdanie, and L. Hakim, "Penerapan React JS dan Axios untuk Pengembangan Front-end Aplikasi iCare," *J. Sess. (Software Dev. Digit. Bus. Intell. Comput. Eng.*, vol. 2, no. 2, pp. 40–46, 2024, doi: 10.57203/session.v2i02.2024.40-46.
[28] I. K. Nasution and H. Mulyono, "Analisis Dan Perancangan Sistem Informasi Layanan Peserta Pelatihan Berbasis Web Pada Lembaga Penjaminan Mutu Pendidikan Provinsi Jambi," *J. Manaj. Sist. Inf.*, vol. 4, no. 4, pp. 455–467, 2019.

This is an open access article under the CC–BY-SA license
Terakreditasi SINTA 5 SK :72/E/KPT/2024 Submitted: **17/10/2025**; Accepted: **24/10/2025**; Published: **07/11/2025**