

Implementasi Workmanager Dalam Sistem Task Scheduling Pada CV. Kenongo Wedding Organizer Di Kota Kediri

Krisi Dwi Pratiwi¹, I Made Suartana²

^{1,2} Program Studi Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya

¹krisidwi.21005@mhs.unesa.ac.id

²madesuartana@unesa.ac.id

Abstrak— CV. Kenongo Wedding Organizer masih menggunakan metode manual dalam pengelolaan tugas, yang berisiko menimbulkan miskomunikasi antar tim. Penelitian ini bertujuan membangun sistem task scheduling berbasis mobile dengan menggunakan WorkManager untuk mendukung penjadwalan tugas dan pengiriman notifikasi secara real-time di latar belakang. Sistem dikembangkan menggunakan Flutter dan Firebase, serta mendukung tiga peran pengguna yaitu, Admin, Project Manager (PM), dan Person in Charge (PIC). WorkManager digunakan untuk mengeksekusi tugas di latar belakang meskipun aplikasi ditutup. Fitur utama meliputi pembuatan tugas, pembagian PIC, perubahan status tugas, unggah bukti, dan notifikasi otomatis sesuai dengan waktu tertentu. Hasil pengujian menunjukkan bahwa seluruh fitur berjalan dengan baik, notifikasi dikirim tepat sesuai kondisi tugas, dan performa sistem yang dilakukan menggunakan DevTools menunjukan stabil dengan rata-rata 59 FPS. Sistem ini dapat meningkatkan koordinasi saat proses persiapan dan eksekusi event.

Kata Kunci— : *WorkManager, Task Scheduling, Wedding Organizer*

I. PENDAHULUAN

Pada era Digital 4.0 Perkembangan teknologi saat ini, telah mendorong Perusahaan, dapat dilihat dari perubahan cara kerja yang mulanya manual hingga bertransformasi secara digital guna untuk meningkatkan efisiensi operasional [1]. Perusahaan tersebut salah satunya adalah *Wedding Organizer*, namun dilansir dari RRI.co.id (30 Agustus 2024) [2] menunjukan bahwa masih banyak *Wedding Organizer* yang kurang menyadari pentingnya koordinasi antar tim, sehingga hal tersebut berdampak terhadap kelancaran acara. Dalam konteks *Wedding Organizer*, koordinasi tim yang baik sangat bergantung pada pengelolaan jadwal dan pembagian tugas yang jelas.

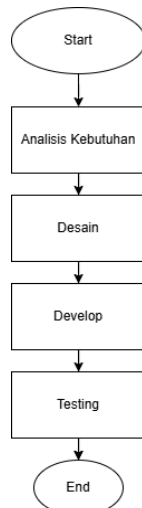
Sebelum mengembangkan sistem, penting untuk memahami proses bisnis yang diterapkan pada setiap Perusahaan, karena proses bisnis merupakan aktivitas yang saling berkaitan yang berfungsi untuk mendukung pencapaian hasil akhir organisasi tersebut [3]. Proses bisnis pada *Wedding Organizer* meliputi empat tahapan utama, yang pertama adalah perencanaan seperti penentuan konsep sesuai kebutuhan Client, tahap kedua persiapan seperti pemilihan vendor dan pengaturan jadwal, tahap ketiga adalah eksekusi, yaitu pelaksanaan hari acara dengan memastikan kelancaran dan koordinasi antar pihak terkait, dan yang terakhir adalah evaluasi, yang dilakukan untuk meminta umpan balik dari Client di *Wedding Organizer* tersebut [4]. Sistem yang dikembangkan dalam penelitian ini berfokus pada tahap persiapan dan eksekusi khususnya dalam penjadwalan tugas dan pengiriman notifikasi secara real-time. Karena Menurut [5] dengan adanya penjadwalan maka akan meningkatkan kedisiplinan dan keprofesionalisme karyawan serta mengurangi risiko miss communication akibat jadwal yang tidak terorganisir dengan baik. Selain itu, dengan adanya sistem notifikasi real-time memastikan bahwa setiap pihak menerima informasi tugas dengan cepat sehingga dapat mempercepat waktu dalam pengambilan keputusan dan juga dalam pelaksanaan tugas [6]. Untuk mendukung hal tersebut, menurut peneliti dibutuhkan sistem Task Scheduling dengan fitur notifikasi untuk memastikan informasi dapat disampaikan secara cepat dan akurat serta meminimalisir risiko miskomunikasi dalam pelaksanaan acara.

Sebagai dasar untuk pengembangan sistem ini Penelitian terdahulu menjadi landasan penting bagi peneliti dalam mengembangkan sistem task scheduling dengan menggunakan WorkManager. Menurut penelitian [7] menjelaskan bahwa mampu mengirimkan notifikasi real-time, guna mempermudah karyawan mengetahui jadwal kerja, serta meningkatkan efisiensi manajemen shift dengan sistem penjadwalan berbasis website yang menggunakan firebase cloud messaging. Penelitian [8] Penelitian ini mengembangkan aplikasi jadwal olahraga outdoor berbasis perangkat bergerak dengan

mempertimbangkan waktu dan kondisi cuaca. Aplikasi ini menggunakan AlarmManager untuk memberikan notifikasi sebagai pengingat pengguna dan memperbarui informasi cuaca secara berkala. Pada penelitian [9] mengembangkan aplikasi manajemen berbasis Android untuk membantu perusahaan *Wedding Organizer* dalam memantau persiapan acara secara terorganisir. Selain itu pada penelitian [10] menunjukan keunggulan WorkManager dalam meningkatkan efisiensi dan memastikan penyelesaian tugas prioritas pada aplikasi android.

Berdasarkan hasil observasi yang telah dilakukan oleh peneliti bahwa Kenongo *Wedding Organizer* merupakan Salah satu Perusahaan dibidang ini yang masih belum menggunakan teknologi digital dalam pengelolaan tugas dan monitoring. Hal ini berpotensi menyebabkan miss communication yang dapat berdampak pada jalannya acara akibat tidak adanya sistem pengingat yang real-time. Oleh karena itu dibutuhkan sistem task scheduling dengan fitur notifikasi real-time dan adanya notifikasi untuk memastikan tugas tersampaikan dan dikerjakan tepat waktu.

II. METODE PENELITIAN



Gbr 1 Alur Penelitian

Pengembangan sistem task scheduling untuk CV. Kenongo *Wedding Organizer* berbasis mobile menggunakan metode WorkManager melalui beberapa tahapan yaitu, analisis kebutuhan, desain, develop, dan juga testing dapat dilihat pada flowchart yang ada pada Error! Reference source not found..

A. Analisis Kebutuhan

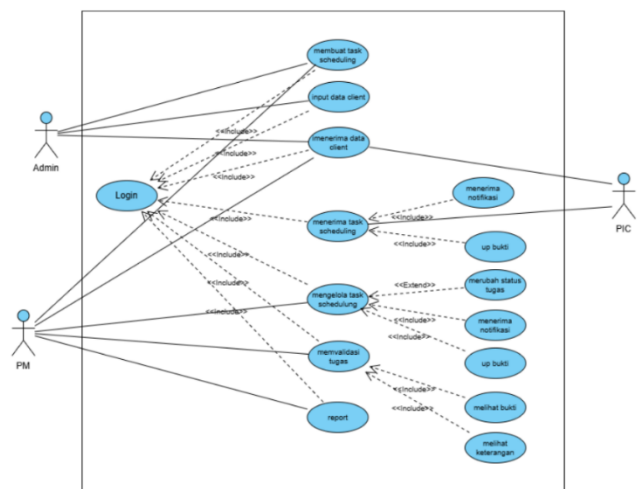
Dalam sistem ini, pembagian kerja dilakukan berdasarkan peran masing-masing PIC seperti dekorasi, registrasi, catering, dan sebagainya, di mana setiap tugas memiliki waktu pelaksanaan yang berbeda. Karena perbedaan waktu ini, diperlukan pengingat tugas yang dikirimkan secara otomatis dan sesuai waktu masing-

masing tugas.. Pada tahap ini dilakukan analisis kebutuhan untuk merancang sistem penjadwalan Task Scheduling menggunakan WorkManager. Identifikasi ini dilakukan dengan mengumpulkan informasi dari pengguna yang akan menjalankan aplikasi tersebut, yaitu Project Manager, Admin dan PIC. Pengumpulan data yang dilakukan dengan metode wawancara. WorkManager dipilih sebagai metode utama karena memungkinkan eksekusi tugas di latar belakang, dan juga memiliki fitur “CallBackDispatcher” yang setiap tugas dijalankan sesuai dengan jadwal yang ada dan memunculkan notifikasi sesuai jadwal atau interval yang ditentukan. Sehingga cocok untuk pengelolaan tugas di *Wedding Organizer*. Dalam pengembangan sistem, selalu terdapat pengguna sistem.

B. Desain

1). Use Case

User dari sistem ini terdapat 3 Aktor Utama, yaitu ada Admin yang selaku untuk menginputkan tugas yang ada beserta data client. *Project Manager* yang memonitoring dan juga membantu PIC dalam tugas tersebut, memyetujui tugas yang akan didelegasikan dan yang sudah di kerjakan. *PIC (Personal In Charge)* yang mengerjakan tugas setelah didelegasikan oleh admin dan disetujui oleh Project Manager, untuk detail pembagiannya dapat dilihat pada Gbr 2

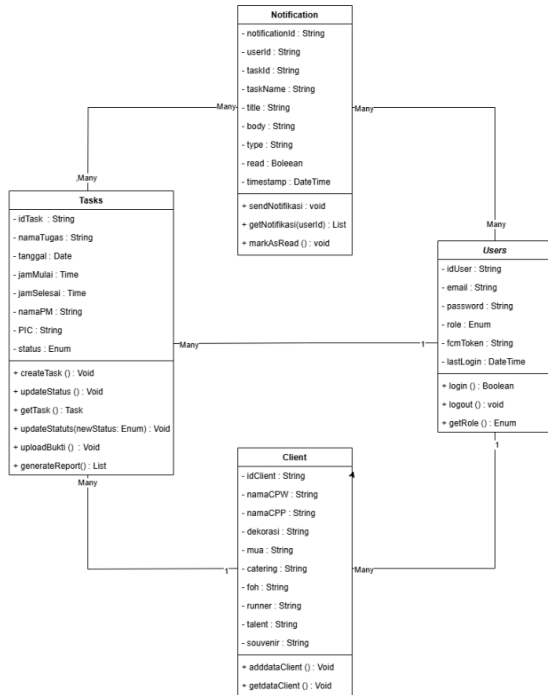


Gbr 2 Use Case

2). Class diagram

Kebutuhan sistem dianalisis melalui pemodelan use case untuk memperoleh gambaran awal mengenai fungsi-fungsi utama yang dibutuhkan oleh pengguna. Berdasarkan hasil tersebut, dilakukan perancangan struktur sistem secara lebih terperinci menggunakan class diagram. Class diagram merupakan salah satu jenis diagram dalam Unified Modeling Language (UML) yang digunakan untuk

memodelkan struktur sistem dalam bentuk kelas-kelas beserta atribut dan metode (methods) yang dimilikinya. Pada Gbr 3 Diagram ini tidak hanya menggambarkan hubungan antar kelas, tetapi juga menunjukkan tanggung jawab dan interaksi antar komponen dalam sistem, sehingga membantu dalam memahami arsitektur logis dari perangkat lunak yang dikembangkan.



Gbr 3 Class Diagram

Pada Gbr 3 merupakan Entity Relationship Diagram yang menggambarkan database dalam sistem task scheduling dan memiliki empat entitas utama yaitu users, task, client dan notification. Setiap user dapat memiliki banyak task dan menerima banyak notifikasi. Task ini berisi informasi seperti nama tugas, waktu pelaksanaan, status dan relasi ke client. Jadi setiap client itu dapat memiliki banyak tugas, hanya satu client aktif yang bisa dibuat per user setelah seluruh tugas sebelumnya selesai. Diagram ini mencerminkan alur sistem dimana tugas dan data client saling terhubung

3). Desain UI

Selain dari segi struktur logis sistem, perancangan juga dilakukan pada aspek tampilan antarmuka pengguna (User Interface). Desain UI merupakan representasi visual dari antarmuka yang menggambarkan elemen-elemen interaktif dalam sistem. Perancangan ini bertujuan agar pengguna dapat dengan mudah memahami dan mengoperasikan aplikasi sesuai dengan perannya masing-masing. Desain UI tidak hanya menitikberatkan pada aspek estetika, tetapi juga mengedepankan bagaimana setiap

elemen mendukung interaksi pengguna. Dalam konteks ini, desain UI diwujudkan dalam bentuk wireframe sebagai gambaran awal struktur dan tata letak komponen antarmuka sebelum tahap implementasi. Terdapat beberapa *Wireframe* pada Gbr 4 antara lain *Home Page*, *Create Task*, *Task*, *Data Client*. Untuk tahap ini dengan *form-form* yang sudah disesuaikan dengan kebutuhan pengguna. Berikut ini adalah komponen utama dari desain UI yang digunakan dalam sistem.



Gbr 4 Wireframe

4). WorkManager

WorkManager merupakan library Android yang digunakan untuk menjalankan tugas latar belakang, bahkan ketika aplikasi ditutup [11]. Dalam sistem Task Scheduling pada CV. Kenongo *Wedding Organizer*, WorkManager ini digunakan untuk memastikan pengiriman notifikasi kepada PIC (Person in Charge) dan Project Manager setiap kali Admin membuat tugas. WorkManager dipilih karena dapat mengelola pengiriman notifikasi meskipun aplikasi tidak sedang aktif. Proses WorkManager disini yaitu :

a. Proses

Admin akan membuat tugas baru melalui form “Create New Task” yang terdiri dari beberapa input, seperti yang ada pada Tabel I:

Tabel I Create New Task

Input	Fungsi
Name Task	Nama tugas yang akan diberikan
Date	Tanggal pelaksanaan tugas
Starting and ending time	Waktu mulai dan selesai tugas
PM	Nama Project Manager
PIC yang bertugas	Orang yang bertanggung jawab untuk menyelesaikan tugas, serta nama orang tersebut.

b. Penggunaan WorkManager

Dalam penggunaan WorkManager ini setelah admin mengisi form “Creat New Task” dan tugas telah berhasil disimpan kedalam database, maka sistem akan memicu WorkManager untuk mengirimkan notifikasi kepada PIC dan Project Manager. Proses ini dilakukan dengan menggunakan presisten langsung “callbackDispatcher” yang berarti pekerjaan ini hanya akan di jalankan sekali tepat setelah tugas baru disimpan. Proses ini memastikan bahwa notifikasi dikirimkan secara tepat waktu, meskipun aplikasi tidak sedang aktif. Notifikasi yang dikirimkan akan berisi informasi mengenai tugas yang telah dibuat. Proses pengiriman notifikasi ini berjalan secara otomatis, yang dapat digunakan untuk kelancaran komunikasi antar admin, PIC, dan Project Manager.

c. Hasil

Hasil implementasi WorkManager dalam sistem Task Scheduling ini adalah pengiriman notifikasi yang real-time dan tepat waktu kepada PIC dan Project Manager. Ketika notifikasi diterima, pengguna dapat langsung mengklik notifikasi tersebut yang akan mengarahkan mereka ke halaman task. Halaman task menampilkan rincian tugas yang meliputi nama tugas, tanggal, waktu pelaksanaan, PIC yang bertugas, serta status tugas yang dapat diubah (Not Completed, Done, Pending).

Selain itu, WorkManager memastikan bahwa notifikasi tetap muncul bahkan ketika aplikasi berjalan di latar belakang atau di tutup. Dengan proses ini, pengguna dapat langsung mengetahui tugas yang diberikan, memulai pekerjaan sesuai instruksi, dan mengupdate status tugas secara real-time. Implementasi WorkManager memastikan bahwa setiap tugas tidak terlewat atau tertunda, dan membantu memastikan keberhasilan penyelesaian tugas tepat waktu.

C. Development

Proses WorkManager dimulai dengan membuat tugas, lalu menetapkan constrain atau Batasan eksekusi. Selanjutnya ditentukan apakah tugas bersifat periodik. Jika tidak, maka dibuat OneTimeWorkRequest. Jika iya, maka menggunakan PeriodicWorkRequest. Kedua jenis request tersebut kemudian dijadwalkan ke WorkManager untuk dijalankan sesuai ketentuan yang telah dilakukan, serta disesuaikan kembali menggunakan bahasa yang digunakan [12].

Prose diatas diterapkan dalam pengembangan sistem Task Scheduling dengan notifikasi untuk menjawab kebutuhan pengguna di CV. Kenongo *Wedding Organizer*. Sistem ini dirancang untuk membantu dalam pengelolaan tugas bagi admin, Project Manager (PM) dan juga PIC

sehingga dapat digunakan di acara. Dalam pengembangan sistem ini, menggunakan Flutter sebagai Framework utama, Firebase sebagai backend untuk autentikasi dan database, serta WorkManager sebagai pengelolaan penjadwalan tugas dan pengiriman notifikasi otomatis.

III. HASIL DAN PEMBAHASAN

Subbab ini menjelaskan hasil dari setiap proses implementasi yang sudah dijabarkan peneliti pada bab 3 sebelumnya. Yang mana, proses dalam membangun sistem *task scheduling* dengan menggunakan metode *WorkManager* pada CV. Kenongo *Wedding Organizer*.

A. Analisis Kebutuhan

Tahap Analisis Kebutuhan merupakan langkah awal dalam proses pembentukan sistem untuk dapat memberikan sistem sesuai dengan kebutuhan pengguna. Berdasarkan hasil wawancara dengan pihak CV. Kenongo *Wedding Organizer*, diketahui bahwa pengelolaan tugas kepada PIC masih dilakukan secara manual melalui komunikasi lisan atau dokumen PDF yang dibagikan via WhatsApp. Metode ini menimbulkan berbagai kendala, seperti PIC yang lupa tugas, jadwal yang bertabrakan, dan ketiadaan bukti penyelesaian tugas. Selain itu, koordinasi antar anggota tim sering terhambat karena informasi tidak tersampaikan secara merata, serta kurangnya pemantauan real-time terhadap progres tugas.

Sebagai solusi, narasumber mengharapkan adanya sistem terintegrasi yang mampu menampilkan informasi penugasan, status pengerjaan, bukti hasil kerja, serta pengingat otomatis. Sistem ini diharapkan dapat meningkatkan keteraturan, akurasi informasi, dan efektivitas kerja tim dalam menjalankan tugas pada setiap acara.

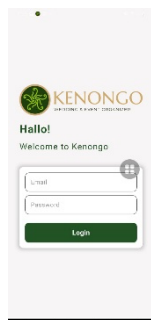
B. Desain

1). Implementasi GUI

Implementasi sistem ini bertujuan untuk memberikan gambaran yang jelas, terstruktur, dan mudah dipahami kepada pengguna mengenai cara kerja sistem serta fungsionalitas yang tersedia. Dengan adanya pemahaman yang baik terhadap alur sistem, pengguna dapat berinteraksi secara baik dan menjalankan perannya sesuai dengan kebutuhan operasional. Penyajian informasi yang informatif dan responsif sangat penting untuk memastikan bahwa setiap tindakan atau masukan dari pengguna dapat direspons dengan akurat oleh sistem, sehingga mendukung kelancaran proses kerja secara keseluruhan..

a. Halaman Login

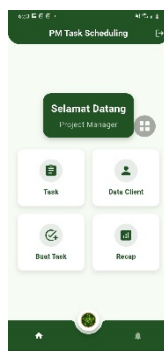
Halaman Login seperti pada Gbr 5 ini menyediakan dua field input yaitu untuk nama dan password, yang digunakan untuk masuk ke dalam sistem sesuai dengan role masing-masing pengguna, seperti Admin, PIC, atau Project Manager. Pada halaman ini juga menentukan tugas dari masing-masing role tersebut, dikarenakan memang terdapat perbedaan tugas dari ketiga role tersebut. Setelah pengguna mengisi kedua data dengan benar dan menekan tombol "Login", sistem akan memanggil fungsi autentikasi yang terdapat pada file `auth_service.dart` untuk memverifikasi kredensial yang dimasukkan. Jika data valid, pengguna akan diarahkan ke halaman sesuai perannya. Jika gagal, maka akan diberikan peringatan otomatis dari sistem.



Gbr 5 Login

b. Home Page

Halaman pada Gbr 6 dirancang untuk menampilkan beberapa menu utama yang dapat diakses sesuai dengan peran pengguna. Pada gambar tersebut, tampilan *Home Page* ditujukan untuk role *Project Manager*, sehingga menu yang ditampilkan telah disesuaikan dengan fungsi dan tanggung jawab yang dimiliki oleh pengguna dengan role tersebut.

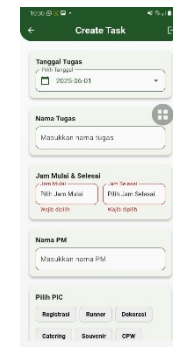


Gbr 6 Home Page

c. Create Task

Menu *create task*, baik di PM maupun di admin, berisikan pemilihan tanggal untuk task akan di lakukan, nama tugas yang akan didelegasikan, jam mulai hingga jam berakhir, nama Project Manager, PIC yang bertugas dan

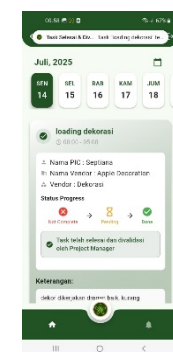
juga yang terakhir Vendor untuk tugas tersebut. Dapat dilihat



Gbr 7 Create Task

d. Task

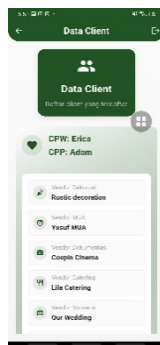
Pada menu *task* pada Gbr 8, digunakan untuk menampilkan hasil *task* yang sudah di buat oleh admin. Halaman ini diambil dari halaman PIC. *User* PIC tidak bisa merubah status apapun, karena langsung otomatis dari sistem. Untuk tugas yang pertama kali didelegasikan akan berstatus "Not Complete", ketika user berhasil mengisi keterangan, dan mengunggah bukti, maka status akan berubah menjadi "Pending", jika tugas tersebut sudah berhasil disetujui oleh *Project Manager*, maka pada halaman PIC status akan berubah menjadi "Done".



Gbr 8 Task

e. Data Client

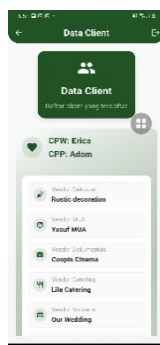
Pada halaman data client pada Gbr 9 berfungsi untuk melihat hasil data client yang sudah ditambahkan oleh admin. Data diambil dari database yang prosesnya membutuhkan waktu. Setelah data berhasil diambil maka disimpan ke variable `_realClient` dan UI di perbarui lewat `setState()` agar dapat tampil dilayar.



Gbr 9 Data Client

f. Recap

Halaman recap pada Gbr 10 ini hanya terdapat di Project Manager saja karena digunakan untuk memudahkan pemantauan progress secara keseluruhan dan dapat digunakan untuk bahan evaluasi. Pengambilan recap ini berdasarkan range tanggal, dan juga per tanggal yang dilaksanakan.



Gbr 10 Recap

2). WorkManager

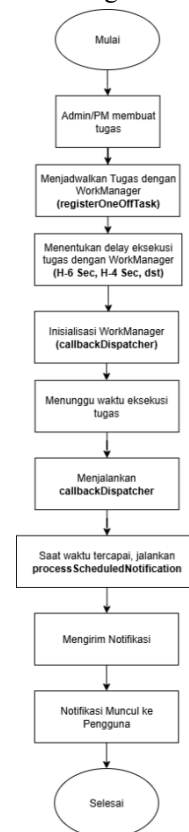
WorkManager merupakan salah satu komponen penting yang digunakan dalam pembentukan *sistem Task Scheduling* pada CV. Kenongo Wedding Organizer. WorkManager memiliki peran penting dalam sistem ini karena memungkinkan sistem untuk menjadwalkan notifikasi tugas bahkan ketika aplikasi dilatar belakang, tetap berjalan. Selain itu juga digunakan untuk memberikan interval notifikasi yang sudah di tentukan seperti H-6 menit, H-4 menit, H-2 menit, dan hingga waktu dikerjakan.

Workmanager memungkinkan aplikasi ini berjalan di latar belakang, bahkan Ketika aplikasi tidak aktif [11]. Dalam konteks Wedding Organizer yang sangat tergantung pada waktu dan koordinasi antar kru, kemampuan WorkManager dalam mengirim notifikasi secara tepat waktu dan otomatis menjadi sangat krusial. Tanpa WorkManager, sistem hanya mengandalkan notifikasi manual yang dapat menyebabkan keterlambatan tugas dan disorganisasi. Fitur ini memastikan tugas tetap dikirim tepat

waktu kepada PIC dan Project Manager. Untuk penggunaan workmanager sendiri sebagai berikut.

a. Flowchart WorkManager

Alur implementasi WorkManager pada Gbr 11 dimulai saat Admin atau Project Manager (PM) membuat tugas untuk PIC. Tugas tersebut kemudian dijadwalkan menggunakan metode `registerOneOffTask`, dengan penentuan delay eksekusi sesuai waktu yang dibutuhkan (misalnya H-6 detik, H-4 detik, dan seterusnya). Setelah itu, WorkManager diinisialisasi melalui `callbackDispatcher`, lalu sistem menunggu hingga waktu eksekusi tiba. Ketika waktu yang dijadwalkan tercapai, sistem menjalankan `callbackDispatcher` yang kemudian memicu fungsi `processScheduledNotification`. Fungsi ini akan mengirimkan notifikasi kepada pengguna terkait. Notifikasi yang muncul berfungsi sebagai pengingat tugas.



Gbr 11 Flowchart WorkManager

b. Create Task

Penggunaan WorkManager untuk mengirimkan notifikasi pada aplikasi ini dimulai Ketika admin telah berhasil melakukan create task dengan form-form yang telah di sediakan terkait pelaksanaan tugas, untuk form yang tersedia seperti yang ada pada Gbr 7

c. Penggunaan WorkManager

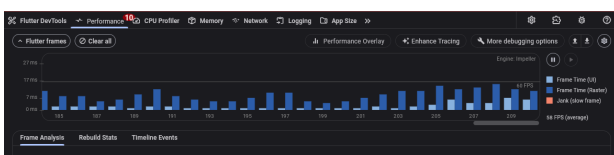
Proses Create Task dilakukan oleh admin melalui form input, sistem perlu menjadwalkan pengingat otomatis agar PIC mendapatkan notifikasi sebelum tugas dimulai. Pada sistem ini WorkManager digunakan untuk Menjalankan fungsi callback secara otomatis pada waktu yang telah ditentukan sebelum tugas di jadwalkan, mengirim notifikasi pengingat kepada PIC yang bertanggung jawab atas tugas tersebut, menjamin bahwa notifikasi tetap terkirim meskipun aplikasi sedang tidak dibuka, menangani banyak tugas sekaligus (Multiple Tasks), menangani tugas dengan delay yang berbeda, mendaftarkan dan menjalankan tugas background secara konsisten

C. Testing

1). Devtools

Pengujian kinerja aplikasi dilakukan dengan menggunakan Flutter DevTools untuk mengetahui performancenya. Berdasarkan hasil pengamatan pada Gbr 12, rata-rata FPS (Frame Per Second) yang diperoleh adalah 59 FPS, yang berarti sangat mendekati angka idealnya adalah 60 FPS. Capaian ini menunjukan bahwa aplikasi berjalan cukup optimal. Menurunnya FPS sedikit ini bisa disebabkan oleh proses pengambilan dan pemantauan data secara real-time dari Firebase, dan disini menggunakan database secara local.

Karena menurut [13] Jika FPS berada pada angka 58 menunjukkan bahwa aplikasi sudah cukup baik, dan hal tersebut bisa terjadi karena aplikasi sering mengakses database secara intensif, seperti menjalankan banyak query, sehingga membebani kinerja perangkat, dan penggunaan database local (localhost) juga bisa membuat sistem bekerja lebih berat, karena aplikasi dan database memakai sumber daya yang sama.



2). Fungsional Testing

Berdasarkan hasil pengujian fungsional yang telah dilakukan terhadap sistem task scheduling dan dapat dilihat pada Tabel II, bahwa seluruh skenario pengujian berhasil dijalankan dengan baik dan sesuai dengan kebutuhan sistem. Fitur-fitur utama seperti penambahan tugas oleh Admin atau Project Manager, tampilan tugas di halaman PM dan PIC, perubahan status tugas, serta pengelolaan data client dapat berfungsi dengan baik tanpa kendala. Selain itu, implementasi WorkManager juga terbukti berjalan sesuai

harapan, di mana notifikasi tetap muncul meskipun aplikasi berada dalam keadaan tertutup, pengingat tugas muncul sesuai waktu yang ditentukan, serta sinkronisasi data berjalan secara otomatis di latar belakang. Dengan demikian, sistem dinilai telah memenuhi fungsionalitas yang dirancang dan layak digunakan dalam operasional CV. Kenongo Wedding Organizer.

Tabel II. Fungsional Testing

Fungsional Testing			
No	Skenario Pengujian	Output	Hasil
1.	Admin atau PM menambahkan tugas	Data tersimpan di Firebase	
2.	Tugas muncul di halaman PM	Tugas yang ditambahkan muncul di daftar	
3.	Tugas muncul di halaman PIC	Tugas yang sesuai muncul di PIC terkait	
4.	Status tugas dapat diubah oleh PM	Status berubah sesuai input	
5.	Tugas divalidasi oleh project manager	Tugas divalidasi sesuai dengan kondisi dan status berubah	

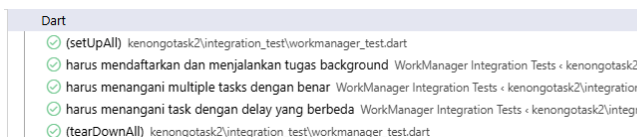
3). Pengujian Integrasi WorkManager

Pengujian integrasi ini dilakukan untuk memastikan bahwa WorkManager dalam sistem berjalan sesuai dengan fungsinya. Pada pengujian ini, file `workmanager_test.dart` digunakan untuk menguji proses penjadwalan tugas latar belakang yang didaftarkan oleh sistem menggunakan metode `registerOneOffTask`. Selain itu, pengujian ini juga mengevaluasi apakah WorkManager mampu menangani multiple task, delay yang berbeda, serta menjadwalkan notifikasi sesuai interval yang di tentukan. Hasil pengujian ditampilkan melalui konsol dan tab Test Result di Visual Studio Code, dan mencakup verifikasi bahwa tidak ada pemanggilan metode yang tidak sesuai. Pengujian ini memiliki Test Case pada tabel III

Tabel III *Integration Testing WorkManager*

Integration Testing WorkManager				
No	Nama Pengujian	Input Data	Expected Result	Status
1.	Mendaftarkan dan menjalankan tugas background	Task dengan delay 1 detik	SharedPreferences menyimpan true pada key WorkManagerTestResult	Passed
2.	Menangani Multiple Tasks	2 buah task berbeda, delay 1 detik	Semua task berhasil dieksekusi, dan efek samping tersimpan	Passed
3.	Menangani Task dengan Delay yang berbeda	Task Tunggal dengan delay 3 detik	Efek samping workManagerTestResult tersimpan dengan	Passed

Berdasarkan tabel III, bahwa hasil pengujian integrasi workmanager menunjukkan bahwa semua scenario berhasil dijalankan dengan baik. WorkManager berhasil mendaftarkan dan menjalankan tugas background, menangani multiple tasks, serta mengelola tugas dengan delay yang berbeda. Pada Gbr 12 merupakan hasil test case pada *integration testing* workmanager, yang dimana semuanya tercentang dengan benar, sehingga Hal ini membuktikan bahwa fungsi WorkManager berjalan sesuai dengan harapan.



Gbr 12 Hasil Test Case Integration Test

4). Unit Testing pada task scheduler

Unit testing pada tabel IV ini dilakukan untuk memastikan bahwa setiap fungsi dalam task scheduler

bekerja sesuai dengan yang diharapkan. Testing ini dilakukan pada file yang bernama `task_notification_service_test.dart`. Pengujian difokuskan pada skenario pengiriman notifikasi yang mencakup berbagai kondisi, seperti tugas baru, perubahan status, penyelesaian tugas, hingga unggahan bukti. Rangkuman hasil pengujian disajikan dalam tabel berikut untuk menunjukkan input data, hasil yang diharapkan, dan status pengujian.

Tabel IV *Unit Testing task scheduler*

Unit Testing task scheduler				
No	Nama Pengujian	Input Data	Expected Result	Status
1.	Penjadwalan notifikasi pengingat tugas secara berkala	Tugas dengan pengingat tugas secara berkala (reminder)	Notifikasi terkirim sesuai interval yang telah ditentukan	Passed
2.	Notifikasi tidak dijadwalkan untuk tugas lampau	Tanggal tugas yang sudah lewat	Semua task berhasil dieksekusi, dan efek samping tersimpan	Passed
3.	Notifikasi untuk tugas baru	Update status tugas ke "pending" / "done"	Notifikasi status terkirim	Passed
4.	Notifikasi saat tugas dibuat ke admin	Membuat tugas baru oleh PM atau Admin	Admin menerima notifikasi terkait tugas yang dibuat	Passed
5.	Notifikasi tugas membutuhkan persetujuan PM	Submit tugas dengan status pending approval	PM menerima notifikasi untuk menyetujui tugas	Passed
6.	Notifikasi tugas disetujui oleh PM	PM mengubah status tugas menjadi approved	Notifikasi dikirim ke PIC bahwa tugas disetujui	Passed
7.	Notifikasi tugas divalidasi oleh PM	PM menekan tombol validasi pada tugas	Notifikasi berhasil dikirim ke PIC	Passed
8.	Notifikasi tugas ditolak oleh PM	PM menolak tugas	PIC menerima notifikasi bahwa tugas ditolak	Passed
9.	Penjadwalan notifikasi pengingat tugas secara berkala	Tugas dengan pengingat tugas secara berkala (reminder)	Notifikasi terkirim sesuai interval yang telah ditentukan	Passed

Berdasarkan tabel IV, bahwa hasil pengujian pada *task scheduler* menunjukkan bahwa semua scenario berhasil dijalankan dengan baik tanpa adanya error. Hal ini membuktikan bahwa fitur notifikasi dalam task scheduling

berjalan dengan benar di berbagai kondisi. Disini Sistem telah mampu memberikan umpan balik yang sesuai melalui notifikasi.

- ✓ should schedule task reminder notifications with correct intervals Task Reminder Tests < TaskScheduler U
- ✓ should not schedule notifications for past tasks Task Reminder Tests < TaskScheduler Unit Tests < kenongot
- ✓ should send notification for new task Task Event Notification Tests < TaskScheduler Unit Tests < kenongotask
- ✓ should send notification for status change Task Event Notification Tests < TaskScheduler Unit Tests < kenong
- ✓ should send notification for task completion Task Event Notification Tests < TaskScheduler Unit Tests < keno
- ✓ should send notification for evidence upload Task Event Notification Tests < TaskScheduler Unit Tests < kenc
- ✓ should send notification for adding remarks Task Event Notification Tests < TaskScheduler Unit Tests < kenoi
- ✓ should send notification for task created to admin Task Event Notification Tests < TaskScheduler Unit Tests
- ✓ should send notification for task needing PM approval Task Event Notification Tests < TaskScheduler Unit
- ✓ should send notification for task approved by PM Task Event Notification Tests < TaskScheduler Unit Tests
- ✓ should send notification for task needing PM validation Task Event Notification Tests < TaskScheduler Unit
- ✓ should send notification for task validated by PM Task Event Notification Tests < TaskScheduler Unit Tests
- ✓ should send notification for task rejected by PM Task Event Notification Tests < TaskScheduler Unit Tests < i
- ✓ should send notification for task rekap Task Event Notification Tests < TaskScheduler Unit Tests < kenongota:

Gbr 13 Hasil Test Case task scheduler

Pada Gbr 13 merupakan hasil *test case task scheduler*, yang dimana semuanya tercentang dengan benar, sehingga hal ini membuktikan bahwa fungsi WorkManager berjalan sesuai dengan harapan.

IV. KESIMPULAN

Berdasarkan hasil implementasi dan pengujian yang dilakukan, sistem task scheduling berbasis mobile dengan menggunakan WorkManager, berhasil dibangun untuk mendukung kelancaran kerja pada CV. Kenongo Wedding Organizer. Sistem ini memfasilitasi 3 jenis pengguna utama yaitu admin, PIC dan juga Project Manager dengan fitur-fitur sesuai dengan role masing-masing seperti pembuatan tugas, penerimaan notifikasi tugas, serta pemantauan status tugas secara real-time dan dapat melihat hasil recap untuk bahan evaluasi.

Pada pengujian unit dan integrasi menunjukkan bahwa sistem mampu mengirim notifikasi sesuai kondisi tugas, seperti tugas baru, perubahan status tugas, selesai, unggah bukti. Serta mampu menjalankan proses di latar belakang, memberikan notifikasi melalui beberapa interval, dan menangani banyak tugas sekaligus melalui WorkManager secara konsisten dan akurat. Dari sisi performa, pengujian menggunakan Flutter DevTools menunjukan bahwa aplikasi berjalan secara optimal dengan rata-rata FPS 59 FPS, yang mendekati angka ideal 60 FPS. Dengan demikian sistem ini layak digunakan dalam mendukung kegiatan persiapan dan eksekusi event yang dilakukan oleh CV. Kenongo Wedding Organizer..

DAFTAR PUSTAKA

- [1] M. Fazis and Tugiah, "Perencanaan Proyek dan Penjadwalan Proyek," vol. 2, no. 12, pp. 1365–1377, 2022.
- [2] Y. P. A. Widodo, "Pentingnya Koordinasi Dan Suksesnya Sebuah Acara," 2024, 2024.
- [3] Y. M. Maulana, "Model Perencanaan Proses Bisnis Berdasarkan Business Process Management Pada Universitas Dinamika," vol. 17, no. 1, pp. 73–85, 2023.
- [4] M. Ariandi and M. I. Ernanto, "Rancang Bangun Wedding Organizer

- Menggunakan Metode Pendekatan User Centered Design," vol. 4, no. 1, pp. 83–93, 2022, doi: 10.47065/josh.v4i1.2249.
- [5] R. Hartono and A. Zein, "PENERAPAN ALGORITMA GENETIKA DAN JARINGAN SYARAF TIRUAN DALAM PENJADWALAN MATA KULIAH Studi Kasus : Prodi Sistem Informasi Universitas Pamulang," vol. VI, no. 03, pp. 6–10, 2023.
- [6] 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," (JTIUST), vol. 09, pp. 284–300, 2024.
- [7] E. Kurniawan and B. S. Wicaksono, "Implementasi Push Notification Pada Sistem Penjadwalan Shift Kerja Dengan Metode Firebase Cloud Messaging Berbasis Website (Studi Kasus : RS Sari Asih Sanghiang)," OKTAL J. Ilmu Komput. dan Sci., vol. 2, no. 3, pp. 890–900, 2023.
- [8] A. Y. Puramandya, A. P. Kharisma, and T. Afirianto, "Pengembangan Aplikasi Perangkat Bergerak Jadwal Olahraga Outdoor berdasarkan Waktu dan Kondisi Cuaca menggunakan Metode Location Based Service , Dark Sky API dan Android Alarm Manager," J. Pengemb. Teknol. Inf. dan Ilmu Komput., vol. 4, no. 2, pp. 581–589, 2020.
- [9] I. W. Prawira and S. R. Nudin, "Rancang Bangun Aplikasi Management Wedding Organizer Berbasis Android," J. Informatics Comput. Sci., pp. 139–150, 2024.
- [10] D. M. Bidkar *et al.*, "Power Restrictions for Android OS : Managing Energy Efficiency and System Performance," Int. J. Comput. Sci. Inf. Technol. Res., vol. 5, no. 4, pp. 1–16, 2024.
- [11] A. Jetpack, "Schedule tasks with WorkManager," 2024. <https://developer.android.com/topic/libraries/architecture/workmanager>.
- [12] Bidushdey, "Understanding Work Manager in Android," medium.com, 2020. <https://medium.com/@bdbidush/understanding-work-manager-in-android-ad0faac00bb>.
- [13] U. Ladayya, D. P. Sari, S. Widodo, N. W. Abdulmajid, and Z. I. Perdana, "Aplikasi mobile sipjuhog (sistem presensi jurnal harian online guru)," JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform., vol. 10, no. 1, pp. 178–200, 2025.