

APLIKASI MUDAH ALIH SEBAGAI PEMBANTU PERIBADI PINTAR UNTUK PRODUKTIVITI HARIAN PELAJAR UNIVERSITI (PEAKFLOW)

AHMAD HAFIY BIN AHMAD FIZRI

ABDUL HADI BIN ABD RAHMAN

*Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia, 43600 UKM Bangi,
Selangor Darul Ehsan, Malaysia*

ABSTRAK

PeakFlow merupakan aplikasi mudah alih yang dibangunkan khusus untuk membantu pelajar universiti menguruskan tugas harian secara sistematik dan meningkatkan produktiviti akademik. Aplikasi ini bertujuan menangani isu pengurusan masa yang lemah, tekanan akademik, dan ketiadaan sistem yang memberikan cadangan tugas secara diperibadikan. Berbeza dengan aplikasi sedia ada yang hanya menyediakan fungsi asas, PeakFlow memperkenalkan dua ciri utama berasaskan kecerdasan buatan, cadangan pintar dan pengoptimuman jadual. Fungsi cadangan pintar membolehkan pelajar berinteraksi melalui teks untuk mendapatkan cadangan tugas baharu yang dihasilkan oleh Model Bahasa Besar (LLM) menerusi OpenRouter API. Manakala fungsi pengoptimuman jadual menjana semula jadual tugas secara automatik berdasarkan keutamaan dan tarikh akhir yang ditetapkan, disertakan dengan penjadualan notifikasi masa nyata. Aplikasi ini dibangunkan menggunakan pendekatan Model-View-Controller (MVC), bahasa pengaturcaraan Kotlin, serta Jetpack Compose untuk antara muka pengguna. Firebase Firestore digunakan sebagai pangkalan data masa nyata, dan WorkManager digunakan untuk pelaksanaan peringatan automatik. Projek ini menunjukkan bagaimana teknologi AI dapat dimanfaatkan untuk menyokong pengurusan tugas pelajar secara lebih pintar, tersusun dan diperibadikan, sejajar dengan inisiatif Revolusi Industri Keempat (4IR).

PENGENALAN

Dalam era digital yang semakin pesat membangun, pelajar universiti sering berdepan dengan cabaran dalam menguruskan tugas akademik, aktiviti kokurikulum, dan kehidupan peribadi mereka secara seimbang. Ketidakupayaan untuk menyusun keutamaan tugas, pengurusan masa yang lemah, serta tekanan akademik yang tinggi merupakan antara faktor utama yang menyumbang kepada penurunan prestasi dan kesejahteraan mental pelajar. Kajian terdahulu menunjukkan bahawa aspek pembelajaran merupakan penyumbang terbesar kepada tekanan dalam kalangan pelajar universiti, lebih tinggi berbanding isu kewangan dan peribadi.

Walaupun terdapat pelbagai aplikasi pengurusan tugas di pasaran, kebanyakannya bersifat generik dan tidak mengambil kira keperluan unik pelajar universiti. Ciri-ciri seperti input manual sepenuhnya, ketiadaan cadangan pintar yang diperibadikan, serta kekurangan fungsi pengurusan masa automatik menyebabkan aplikasi sedia ada kurang membantu dalam menangani cabaran sebenar pelajar. Tambahan pula, interaksi antara pengguna dan sistem sering kali tidak fleksibel serta tidak disesuaikan dengan konteks akademik.

Sehubungan itu, satu aplikasi mudah alih bernama PeakFlow telah dibangunkan sebagai penyelesaian pintar untuk menyokong produktiviti harian pelajar universiti. Aplikasi ini menggabungkan teknologi kecerdasan buatan (AI) dan Model Bahasa Besar (LLM) bagi menyediakan fungsi seperti cadangan tugas pintar, pengoptimuman jadual, dan peringatan masa nyata. Pengguna boleh berinteraksi dengan aplikasi secara teks bagi menerima cadangan tugas yang diperibadikan mengikut corak kerja dan keutamaan masing-masing.

Dengan menggunakan teknologi seperti Jetpack Compose, Firebase Firestore, dan OpenRouter API, serta dibangunkan berdasarkan pendekatan seni bina Model-View-Controller (MVC), aplikasi PeakFlow menyasarkan untuk menjadi pembantu peribadi pintar yang mampu membantu pelajar bukan sahaja dalam mengatur tugas tetapi juga mengekalkan keseimbangan kehidupan akademik dan peribadi. Projek ini sejajar dengan inisiatif Revolusi Industri Keempat (4IR) yang menekankan penggunaan teknologi pintar dan data masa nyata dalam penyelesaian kehidupan harian.

KAJIAN LITERATUR

Dalam membangunkan aplikasi PeakFlow, kajian literatur telah dijalankan untuk mengenal pasti amalan terbaik dalam pengurusan tugas serta potensi penggunaan teknologi kecerdasan buatan (AI) dalam aplikasi mudah alih. Tujuan utama kajian ini adalah untuk memahami kekuatan dan kelemahan sistem sedia ada, serta mengenal pasti ruang inovasi yang boleh dimanfaatkan bagi meningkatkan produktiviti harian pelajar universiti melalui pendekatan yang lebih pintar dan diperibadikan.

Kajian mendapati bahawa pelajar universiti berdepan dengan cabaran besar dalam mengurus masa, mengatur keutamaan tugas, dan mengekalkan keseimbangan antara kehidupan akademik dan peribadi. Kajian oleh Ahmad dan Abdul Latif (2023) menunjukkan bahawa tekanan akademik yang tinggi merupakan faktor utama yang menyumbang kepada kebimbangan dalam kalangan pelajar universiti. Tambahan pula, Li et al. (2022) mencadangkan bahawa penerapan AI dalam aplikasi mudah alih dapat meningkatkan keberkesanan sistem melalui cadangan dan interaksi yang diperibadikan.

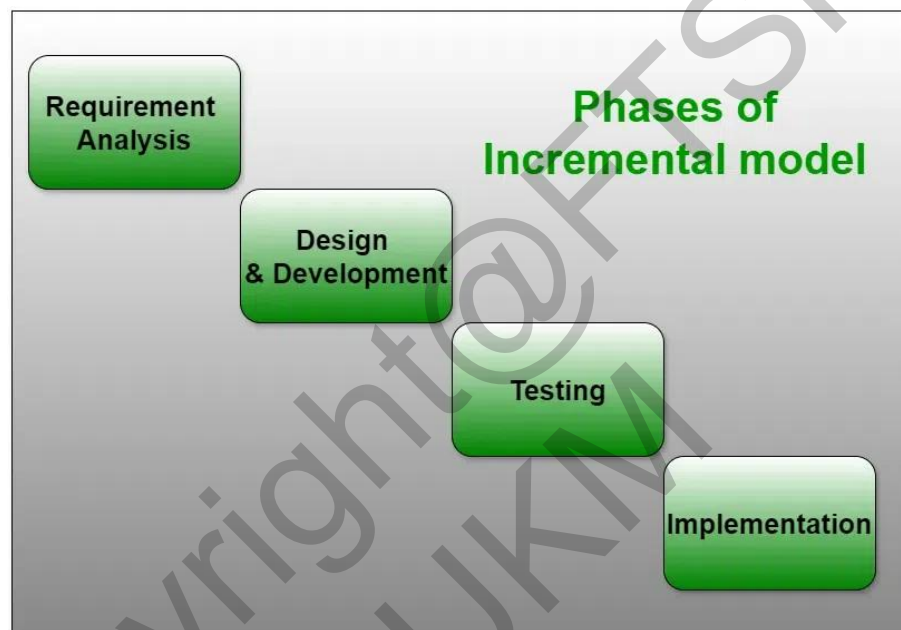
Dari segi teknologi, sistem sedia ada seperti Todoist, TickTick, dan Any.do telah digunakan secara meluas sebagai aplikasi pengurusan tugas. Walau bagaimanapun, kajian terhadap sistem-sistem ini mendedahkan beberapa kekurangan ketara. Todoist, sebagai contoh, memerlukan input manual dan tidak menyediakan cadangan pintar berdasarkan corak kerja pengguna. TickTick pula menawarkan penjadualan visual dan pengingat lokasi, namun masih kekurangan integrasi AI. Any.do menyokong integrasi pihak ketiga dan peringatan pintar, tetapi tidak mempunyai fungsi pemprosesan bahasa semula jadi untuk memahami konteks pengguna.

Berdasarkan pemerhatian ini, dapat disimpulkan bahawa masih terdapat kekosongan dalam penyelesaian sedia ada yang boleh diisi dengan pendekatan yang lebih inovatif. PeakFlow dibangunkan dengan mengambil kira kelemahan ini dan memperkenalkan ciri-ciri seperti cadangan tugas pintar dan pengoptimuman jadual berasaskan AI, yang disokong oleh teknologi Model Bahasa Besar (LLM). Fungsi-fungsi ini membolehkan sistem mengenal pasti tugas yang perlu diberi keutamaan dan menyusun semula jadual harian pelajar secara automatik, selaras dengan prinsip Revolusi Industri Keempat (4IR).

Secara keseluruhannya, kajian literatur ini membuktikan keperluan terhadap aplikasi yang lebih pintar, diperibadikan, dan responsif dalam membantu pelajar universiti menguruskan tugas dengan lebih cekap. PeakFlow tampil sebagai penyelesaian inovatif yang memanfaatkan AI untuk menjawab cabaran ini.

METODOLOGI

Pembangunan aplikasi PeakFlow dilaksanakan berasaskan model pembangunan Inkremental, iaitu pendekatan yang menekankan pelaksanaan sistem secara berperingkat. Melalui kaedah ini, setiap modul dibangunkan secara individu mengikut turutan logik — bermula dengan Analisis Keperluan, diikuti Reka Bentuk dan Pembangunan, Pengujian, dan seterusnya Pelaksanaan ke dalam sistem sebenar. Dengan setiap fasa pembangunan, sistem menjadi semakin lengkap dan stabil, sambil memberi ruang untuk penambahbaikan berterusan berdasarkan hasil pengujian dan maklum balas pengguna.



Rajah 1: Fasa-fasa Metodologi Pembangunan Inkremental

Sumber: <https://www.geeksforgeeks.org/software-engineering/software-engineering-incremental-process-model/>

Fasa pertama pembangunan dimulakan dengan aktiviti pengumpulan dan analisis keperluan, yang bertujuan mengenal pasti masalah utama yang dihadapi oleh pelajar universiti dalam pengurusan tugas harian. Proses ini melibatkan kajian literatur, semakan sistem sedia ada, dan analisis terhadap tingkah laku pengguna sasaran. Dapatan daripada penyelidikan menunjukkan bahawa pelajar sering menghadapi kesukaran dalam menyusun keutamaan, mengurus masa, dan mengekalkan keseimbangan antara tugas akademik dan kehidupan peribadi.

Berdasarkan analisis tersebut, keperluan sistem dibahagikan kepada dua kategori utama: keperluan fungsian dan keperluan bukan fungsian. Keperluan fungsian termasuk keupayaan untuk mendaftar dan log masuk pengguna, menambah serta mengemaskini tugas, menetapkan peringatan, serta menerima cadangan tugas pintar dan penjadualan semula berdasarkan keutamaan. Keperluan bukan fungsian pula merangkumi prestasi sistem, antara muka mesra pengguna, kebolehskalaan aplikasi, dan keselamatan data. Semua keperluan ini dijadikan asas utama dalam reka bentuk sistem yang akan dibangunkan.

Setelah keperluan sistem dikenal pasti dengan jelas, pembangunan diteruskan dengan proses reka bentuk dan pembinaan modul-modul sistem. Aplikasi dibangunkan menggunakan Android Studio dengan bahasa pengaturcaraan Kotlin, manakala antara muka pengguna direka sepenuhnya menggunakan Jetpack Compose, yang menyediakan pendekatan deklaratif untuk membina UI yang moden dan fleksibel.

Struktur sistem menggunakan seni bina Model-View-Controller (MVC) bagi memisahkan tanggungjawab antara lapisan logik, paparan dan data. Ini membolehkan setiap komponen dikendalikan secara modular, sekali gus memudahkan penyelenggaraan dan pengujian.

Komponen utama yang dibangunkan termasuk sistem pengurusan akaun pengguna, pengurusan tugas harian, serta dua fungsi berasaskan AI iaitu cadangan pintar dan pengoptimuman jadual. Kedua-dua fungsi ini menggunakan Large Language Model (LLM) melalui OpenRouter API, di mana aplikasi akan menghantar senarai tugas serta pertanyaan pengguna ke model AI dan menerima respons dalam bentuk cadangan yang bersesuaian atau jadual tugas baharu yang telah dioptimumkan.

Sementara itu, data aplikasi disimpan menggunakan Firebase Firestore, yang menyediakan penyimpanan berasaskan awan dan capaian masa nyata. Fungsi peringatan pula dijalankan menggunakan WorkManager, membolehkan sistem menjadual dan memaparkan notifikasi walaupun aplikasi berada dalam latar atau tidak aktif sepenuhnya.

Pengujian dilaksanakan secara menyeluruh bagi memastikan aplikasi mencapai tahap kefungsian dan kebolehgunaan yang tinggi. Beberapa pendekatan telah diambil dalam pelaksanaan fasa ini.

Pertama, ujian unit dijalankan terhadap fungsi-fungsi individu seperti log masuk, pendaftaran pengguna, pengurusan tugas, dan integrasi dengan API AI. Tujuan ujian ini

adalah untuk mengesan sebarang ralat pada peringkat logik awal pembangunan dan memastikan setiap fungsi beroperasi seperti yang dijangka.

Kedua, aplikasi diuji melalui ujian penerimaan pengguna (User Acceptance Test, UAT) bersama beberapa pengguna sasaran. Mereka diminta untuk mencuba aplikasi dan memberikan maklum balas melalui soal selidik berkaitan kefungsian, kemudahan penggunaan, dan kepuasan keseluruhan.

Ketiga, ujian kebolehgunaan dilaksanakan untuk menilai tahap kefahaman terhadap antara muka pengguna. Antara aspek yang dinilai termasuk kejelasan ikon dan mesej sistem, keselesaan navigasi, dan susun atur paparan. Ujian ini memberi tumpuan kepada pengalaman pengguna secara holistik bagi memastikan aplikasi mudah difahami tanpa memerlukan panduan tambahan.

Kesemua jenis pengujian ini dijalankan secara berulang sepanjang pembangunan dan setiap kali fungsi baharu ditambah, bagi memastikan kestabilan sistem kekal terjamin.

Fasa akhir metodologi ialah pelaksanaan sistem, di mana setiap modul atau fungsi yang telah dibangunkan dan diuji akan diintegrasikan ke dalam versi penuh aplikasi. Integrasi ini dilakukan secara berperingkat bagi mengelakkan gangguan kepada sistem sedia ada. Dengan setiap penambahan modul, sistem diuji semula secara keseluruhan bagi memastikan tiada konflik atau gangguan yang berlaku antara fungsi-fungsi lain.

Sebagai contoh, setelah modul cadangan pintar berjaya dibangunkan, fungsi ini dimasukkan bersama pengurusan tugas dan diuji untuk memastikan cadangan AI dapat disampaikan kepada pengguna tanpa mengganggu paparan tugas biasa. Begitu juga dengan fungsi pengoptimuman jadual dan WorkManager yang dijadualkan secara serentak untuk tugas-tugas baharu yang dijana oleh model AI.

Pendekatan pelaksanaan secara modular dan tersusun ini membolehkan sistem dibangunkan dengan lebih terkawal, sekaligus menghasilkan sebuah aplikasi lengkap yang berfungsi lancar, stabil dan memenuhi keperluan sebenar pengguna sasarannya, iaitu pelajar universiti.

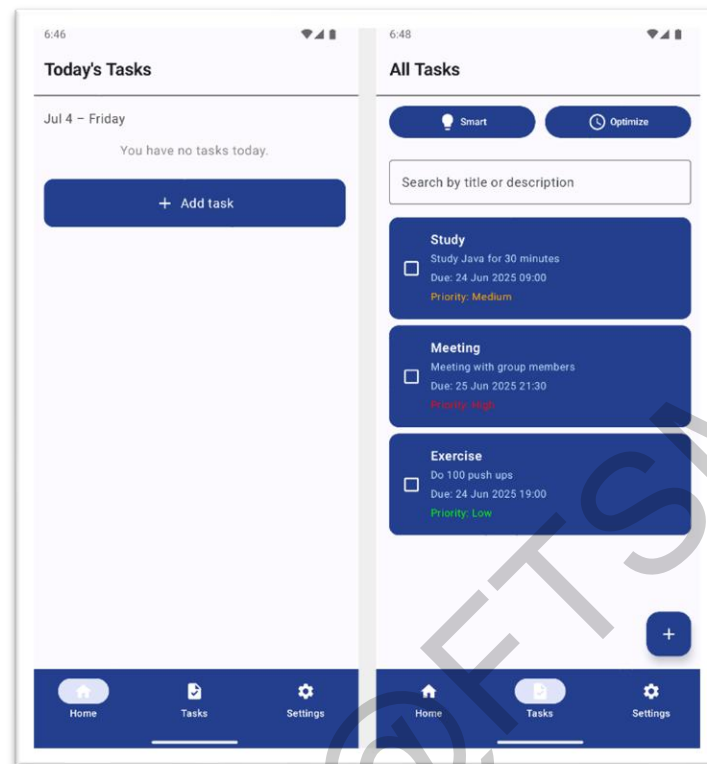
HASIL

Aplikasi PeakFlow telah berjaya dibangunkan sebagai satu penyelesaian mudah alih yang stabil, moden, dan mampu membantu pelajar universiti menguruskan tugas harian mereka dengan lebih tersusun. Pembangunan aplikasi ini merangkumi pelbagai modul yang dibina dan diintegrasikan secara berperingkat, mengikut pendekatan model inkremental. Setiap fungsi diuji secara sistematik dan telah mencapai tahap kefungsiian yang memuaskan berdasarkan keperluan projek.



Rajah 2: Antara muka skrin log masuk dan pendaftaran pengguna

Pengguna memulakan interaksi mereka dengan aplikasi melalui skrin log masuk atau pendaftaran akaun. Maklumat yang diperlukan semasa pendaftaran termasuk nama pengguna, emel dan kata laluan. Aplikasi dilengkapi dengan mekanisme pengesahan input bagi mengelakkan kesilapan seperti emel tidak sah atau kata laluan tidak mencukupi. Sekiranya berlaku ralat, mesej maklum balas dipaparkan kepada pengguna secara terus melalui antaramuka. Fungsi ini menggunakan Firebase Authentication untuk menjamin keselamatan dan kecekapan proses pengesahan pengguna.



Rajah 3: Antara muka halaman utama dan senarai semua tugas

Setelah berjaya log masuk, pengguna akan dibawa ke halaman utama yang memaparkan tugas yang dijadualkan untuk hari semasa. Halaman ini direka bentuk secara ringkas namun efektif, dengan menekankan kebolehbacaan dan kejelasan status tugas. Pengguna juga boleh mengakses halaman *Semua Tugas* yang menyenaraikan semua tugas yang telah dimasukkan dalam sistem. Melalui halaman ini, pengguna boleh menanda tugas sebagai siap, membuat suntingan, atau memadamnya. Fungsi carian turut disediakan untuk memudahkan penapisan tugas mengikut tajuk atau deskripsi, dan setiap perubahan disimpan serta-merta ke Firestore.

The image displays two mobile application screens side-by-side, illustrating the 'Add Task' and 'Edit Task' interfaces. Both screens feature a light purple header with a back arrow and the respective title. A note at the top of each form states 'Fields marked with * are required.'.

Add Task Screen (Left):

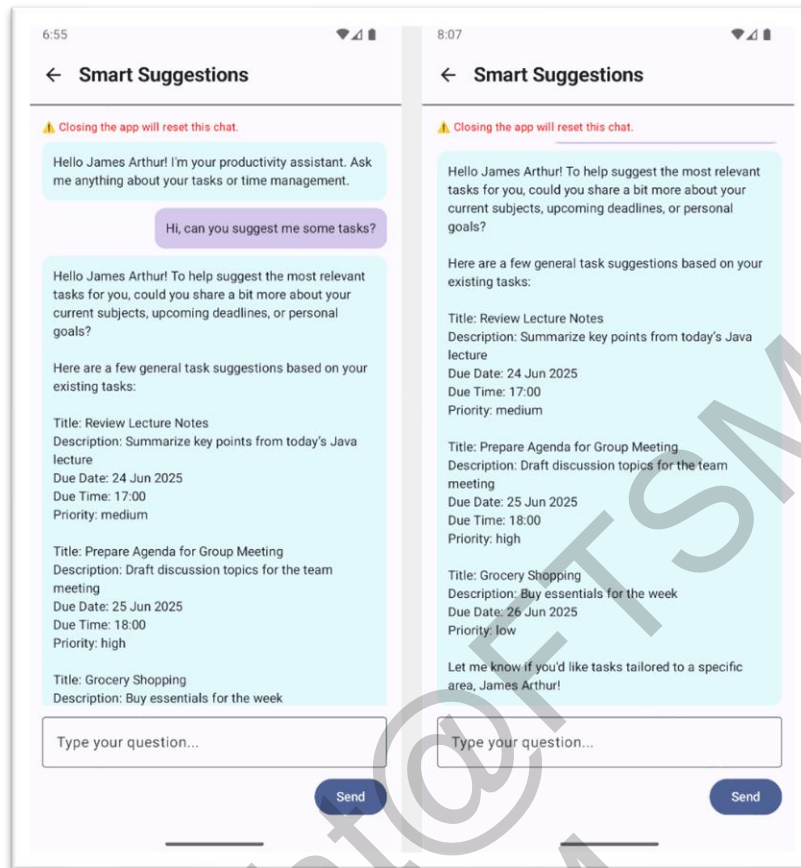
- Title *: Empty text input field.
- Description: Empty text input field.
- Start Date: Empty date picker field.
- Due Date *: 04 Jul 2025 (selected date).
- Start Time: Empty time picker field.
- Due Time: Empty time picker field.
- Reminder before (minutes): 0 (selected value).
- Priority *: medium (selected value).
- Save: Blue button at the bottom.

Edit Task Screen (Right):

- Title *: Meeting (filled text input field).
- Description: Meeting with group members (filled text input field).
- Start Date: Empty date picker field.
- Due Time: 21:30 (selected time).
- Start Time: Empty time picker field.
- Due Date *: 25 Jun 2025 (selected date).
- Priority *: high (selected value).
- Reminder before (minutes): 30 (selected value).
- Save Changes: Blue button at the bottom.

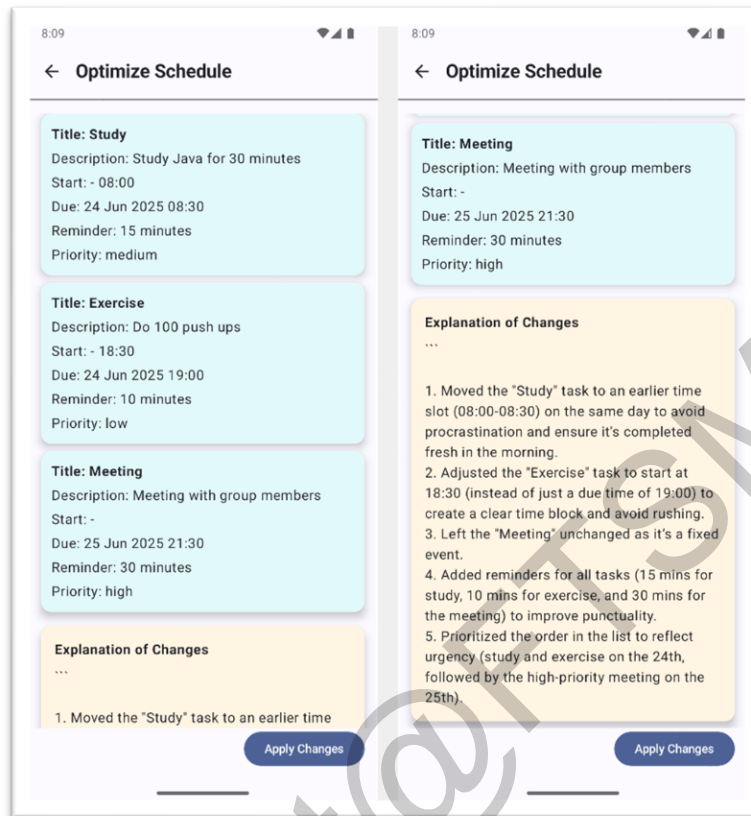
Rajah 4: Antara muka tambah dan ubah tugas

Aplikasi ini juga menyediakan halaman khusus bagi menambah atau mengubah butiran tugas. Antara input yang disokong termasuk tajuk, deskripsi, tarikh mula dan tamat, masa, keutamaan, serta masa peringatan. Pemilihan tarikh dan masa dilakukan melalui dialog yang dibuka apabila pengguna menekan kawasan interaktif. Aplikasi turut menandakan medan wajib dengan asterisk (*) dan memaparkan mesej ralat menggunakan snackbar. Setiap tugas yang disimpan akan ditambahkan ke pangkalan data Firestore, dan jika peringatan ditetapkan, ia akan dijadualkan melalui WorkManager.



Rajah 5: Antara muka fungsi cadangan pintar

Salah satu ciri utama aplikasi ialah fungsi cadangan pintar, yang membolehkan pengguna berinteraksi dengan model LLM melalui antara muka gaya chatbot. Pengguna boleh menaip sebarang soalan berkaitan pengurusan masa atau tugas mereka, dan sistem akan menghantar prompt tersuai ke API OpenRouter. Prompt ini mengandungi mesej pengguna, senarai tugas sedia ada dan arahan sistem yang direka untuk menghasilkan cadangan yang relevan. Respons yang diterima daripada model AI diproses dan dipaparkan dalam bentuk kad cadangan yang kemas, lengkap dengan tajuk, deskripsi, keutamaan dan tarikh akhir yang disarankan.



Rajah 6: Antara muka fungsi pengoptimuman jadual

Fungsi pengoptimuman jadual pula membolehkan pengguna menjana semula jadual tugas yang lebih teratur berdasarkan keutamaan dan tarikh akhir. Apabila fungsi ini diaktifkan, senarai tugas pengguna akan dihantar ke model AI yang kemudian memulangkan susunan baharu yang telah dioptimumkan bersama penjelasan untuk setiap perubahan. Aplikasi akan memaparkan cadangan tersebut dan membenarkan pengguna untuk memilih sama ada ingin menyimpan perubahan tersebut atau tidak. Jika dipersetujui, tugas baharu akan dimasukkan ke dalam Firestore dan dijadualkan dengan peringatan baharu menggunakan WorkManager.

Selain antara muka yang dapat dilihat secara langsung oleh pengguna, aplikasi ini turut menyokong fungsi peringatan masa nyata melalui sistem notifikasi Android. Setiap tugas yang telah ditetapkan dengan masa peringatan akan dijadualkan secara automatik menggunakan WorkManager, yang memastikan notifikasi dipaparkan pada masa yang ditentukan walaupun aplikasi berada dalam keadaan latar atau telah ditutup sepenuhnya. Fungsi ini membolehkan pengguna menerima peringatan tugas penting secara konsisten tanpa perlu membuka aplikasi, sekali gus menyumbang kepada pengalaman penggunaan yang lebih efisien dan tidak mengganggu.

Secara keseluruhannya, aplikasi PeakFlow telah berjaya memenuhi objektif projek dengan menyediakan pelbagai fungsi yang saling melengkapi. Fungsi-fungsi utama bukan sahaja berfungsi dengan baik secara berasingan, tetapi juga berinteraksi secara lancar apabila digabungkan. Keupayaan aplikasi untuk memberikan cadangan tugas dan menyusun jadual secara pintar menjadikannya lebih daripada sekadar aplikasi senarai tugas biasa. Maklum balas awal daripada pengguna ujian menunjukkan aplikasi ini mudah digunakan, bermanfaat dalam merancang tugas, dan berpotensi untuk dikembangkan ke peringkat yang lebih tinggi.

UJIAN PENERIMAAN PENGGUNA

Ujian penerimaan pengguna dilaksanakan bagi memastikan aplikasi PeakFlow memenuhi keperluan fungsian yang telah ditetapkan dalam fasa analisis dan pembangunan. Pengujian ini menilai sejauh mana fungsi-fungsi utama aplikasi dapat digunakan dengan baik oleh pengguna sebenar tanpa sebarang masalah besar, sekaligus mengesahkan kesediaan sistem untuk digunakan dalam konteks sebenar.

Seramai 10 orang responden telah terlibat dalam ujian ini, terdiri daripada pelajar universiti yang mewakili kumpulan pengguna sasaran. Mereka diminta mencuba aplikasi dan kemudian memberikan maklum balas melalui soal selidik yang disediakan. Soal selidik tersebut mengandungi 10 pernyataan berkaitan fungsi aplikasi seperti log masuk, pengurusan tugas, penggunaan ciri AI, dan sistem peringatan. Setiap pernyataan dinilai menggunakan skala Likert lima mata, di mana nilai 1 mewakili Sangat Tidak Setuju dan nilai 5 mewakili Sangat Setuju.

Jadual 1: Keputusan Ujian Penerimaan Pengguna

No.	Soalan	Skor Purata
1	Saya berjaya mendaftar akaun dengan mudah.	4.7
2	Proses log masuk mudah dan tidak bermasalah.	4.8
3	Saya boleh menambah tugas dengan lancar.	4.8
4	Saya mudah menetapkan peringatan tugas.	4.7

No.	Soalan	Skor Purata
5	Fungsi AI memberikan cadangan yang berguna.	4.6
6	Ciri pengoptimuman jadual membantu saya merancang tugas.	4.6
7	Saya menerima notifikasi tugas tepat pada masanya.	4.7
8	Aplikasi ini mudah untuk digunakan dan difahami.	4.8
9	Navigasi antara skrin dalam aplikasi ini adalah lancar.	4.7
10	Saya berpuas hati dengan keseluruhan fungsi dalam aplikasi ini.	4.8

Hasil ujian menunjukkan bahawa fungsi utama aplikasi telah diterima baik oleh pengguna. Secara purata, kesemua pernyataan menerima skor yang tinggi, menandakan bahawa fungsi seperti log masuk, pendaftaran pengguna, tambah dan ubah tugas, serta sistem cadangan pintar dan pengoptimuman jadual berjaya dilaksanakan seperti yang dirancang. Antara aspek yang menerima skor tertinggi ialah kemudahan untuk menggunakan aplikasi tanpa perlu bantuan, ketepatan fungsi sistem, serta kesediaan pengguna untuk menggunakan aplikasi ini dalam kehidupan sebenar mereka.

Maklum balas ini membuktikan bahawa aplikasi PeakFlow bukan sahaja lengkap dari segi fungsi asas, malah telah memenuhi jangkaan dan keperluan pengguna secara menyeluruh.

UJIAN KEBOLEHGUNAAN

Selain daripada memastikan fungsi aplikasi dapat digunakan, satu lagi aspek penting yang diuji ialah kebolehgunaan antara muka sistem. Ujian kebolehgunaan memberi tumpuan kepada tahap kefahaman, kemudahan interaksi, dan pengalaman pengguna secara keseluruhan apabila

menggunakan aplikasi ini. Tujuannya adalah untuk memastikan bahawa aplikasi ini tidak hanya berfungsi, tetapi juga mesra pengguna, mudah difahami dan menarik untuk digunakan.

Ujian ini juga melibatkan 10 orang responden yang sama, yang diminta memberikan maklum balas terhadap 10 pernyataan berkaitan antara muka pengguna. Pernyataan-pernyataan ini menyentuh aspek seperti susun atur aplikasi, kejelasan ikon dan teks, kemudahan navigasi, serta kefahaman terhadap fungsi-fungsi AI yang disediakan. Penilaian diberikan menggunakan skala Likert lima mata, sama seperti sebelumnya, dengan nilai 1 mewakili Sangat Tidak Setuju dan nilai 5 mewakili Sangat Setuju.

Jadual 2: Keputusan Ujian Kebolehgunaan

No	Soalan	Skor Purata
1	Reka bentuk antara muka aplikasi adalah menarik dan moden.	4.7
2	Teks, ikon, dan butang dalam aplikasi mudah difahami dan digunakan.	4.8
3	Saya dapat mencari fungsi yang diperlukan dengan mudah.	4.6
4	Susun atur skrin aplikasi memudahkan saya untuk menavigasi aplikasi.	4.7
5	Saya berasa selesa menggunakan aplikasi ini secara konsisten.	4.8

Hasil keputusan menunjukkan bahawa aplikasi menerima maklum balas yang sangat positif dari segi kebolehgunaan. Semua pernyataan memperoleh skor purata yang tinggi, menunjukkan tahap kepuasan pengguna terhadap antara muka dan pengalaman penggunaan aplikasi. Responden menyatakan bahawa aplikasi mudah digunakan walaupun tanpa panduan, susun atur kemas dan mudah difahami, serta fungsi cadangan pintar dan pengoptimuman jadual dapat difahami dengan jelas walaupun melibatkan interaksi dengan AI.

Terdapat juga beberapa maklum balas membina berkaitan kedudukan ikon dan penambahbaikan pada aspek visual untuk peranti bersaiz kecil, namun isu ini tidak menjejaskan kefungisian aplikasi secara keseluruhan. Secara keseluruhannya, keputusan ujian kebolehgunaan menunjukkan bahawa aplikasi bukan sahaja berfungsi dengan baik, tetapi juga menyampaikan pengalaman penggunaan yang selesa dan sesuai untuk kumpulan pengguna sasaran.

CADANGAN PENAMBAHBAIKAN

Walaupun aplikasi PeakFlow telah berjaya memenuhi objektif pembangunan dan menyediakan fungsi utama yang stabil serta mesra pengguna, terdapat beberapa ciri tambahan yang berpotensi untuk memperkukuh lagi keberkesanan dan pengalaman penggunaan aplikasi ini pada masa hadapan. Antara penambahbaikan yang boleh dipertimbangkan ialah meningkatkan kecekapan masa respons untuk fungsi Smart Suggestions dan Optimize Schedule, terutamanya dengan mengoptimumkan penggunaan model AI atau menerapkan teknik caching untuk permintaan yang serupa.

Selain itu, penambahan fungsi tugas berulang juga boleh memperluas kegunaan aplikasi kepada aktiviti berkala seperti kuiz mingguan, sesi tutorial, atau latihan rutin. Fungsi ini akan menjadikan aplikasi lebih fleksibel dalam menyokong tugas jangka panjang yang bersifat berulang. Aplikasi juga boleh diperbaiki dengan menambah ciri pengesahan sebelum pemadaman atau fungsi “undo” selepas tindakan kritikal dilakukan, bagi mengurangkan risiko kehilangan data akibat kesilapan tidak sengaja.

Dari sudut organisasi tugas, antara muka boleh dipertingkatkan dengan menyokong sistem label atau kategori berwarna bagi membolehkan pengguna menyusun tugas mengikut kursus, jenis kerja, atau tahap keutamaan. Paparan kalendar dalam bentuk mingguan atau bulanan juga boleh disediakan untuk memberikan gambaran yang lebih menyeluruh terhadap tugas yang telah dijadualkan, membolehkan pengguna merancang tugas mereka secara visual dan strategik.

Tambahan lagi, menyokong mod gelap automatik atau pilihan tema manual akan meningkatkan keselesaan visual pengguna serta memberi lebih kawalan terhadap suasana penggunaan aplikasi.

KESIMPULAN

Secara keseluruhannya, pembangunan aplikasi PeakFlow telah berjaya dilaksanakan mengikut perancangan yang ditetapkan dan memenuhi objektif utama projek. Aplikasi ini dibangunkan sebagai pembantu pintar harian yang bertujuan membantu pelajar universiti mengurus tugas dengan lebih cekap dan tersusun. Melalui penggunaan pendekatan model pembangunan inkremental, aplikasi ini telah dibina secara berperingkat dan diuji bagi memastikan kestabilan serta kebolehgunaan setiap modul.

Antara fungsi utama yang berjaya dibangunkan termasuk log masuk pengguna, pengurusan tugas harian, sistem peringatan masa nyata, serta dua ciri berasaskan kecerdasan buatan iaitu cadangan pintar dan pengoptimuman jadual. Integrasi dengan model LLM melalui OpenRouter telah membolehkan sistem memberikan cadangan tugas dan penjadualan yang diperibadikan, selaras dengan keperluan pengguna. Penggunaan Jetpack Compose sebagai teknologi utama untuk antara muka turut menyumbang kepada pengalaman pengguna yang lebih moden, mesra dan konsisten.

Hasil daripada ujian penerimaan pengguna dan pengujian kebolehgunaan menunjukkan bahawa aplikasi ini bukan sahaja berfungsi dengan baik dari segi teknikal, malah mendapat maklum balas positif dari segi kefahaman, kemudahan penggunaan dan reka bentuk antara muka. Keseluruhan fungsi telah dilaksanakan mengikut keperluan, manakala potensi penambahbaikan telah dikenal pasti untuk pembangunan lanjut pada masa hadapan.

Projek ini juga membuktikan bahawa integrasi antara pengurusan tugas tradisional dan teknologi moden seperti LLM dapat menghasilkan penyelesaian yang lebih pintar, bersifat adaptif dan berorientasikan pengguna. Dengan asas teknikal yang kukuh dan tindak balas pengguna yang menggalakkan, aplikasi PeakFlow mempunyai potensi untuk dikembangkan lagi dan mungkin dijadikan alat sokongan sebenar dalam kehidupan akademik harian pelajar.

PENGHARGAAN

Segala puji dan syukur dipanjatkan kepada Allah SWT atas limpahan rahmat, hidayah, serta kekuatan yang diberikan dalam menyelesaikan laporan ilmiah ini. Tanpa izin-Nya, usaha ini tidak akan berjaya dengan sempurna.

Saya ingin merakamkan setinggi-tinggi penghargaan dan terima kasih kepada penyelia saya, Prof. Madya Dr. Abdul Hadi Bin Abd Rahman, yang telah memberi bimbingan, nasihat, dan dorongan sepanjang proses penyelidikan, pembangunan dan penulisan projek ini. Sokongan dan tunjuk ajar beliau amat bermakna dalam memastikan projek ini berada pada tahap yang terbaik.

Ucapan terima kasih juga saya tujukan kepada Fakulti Teknologi dan Sains Maklumat dan Universiti Kebangsaan Malaysia atas segala kemudahan, sokongan, dan sumber yang disediakan sepanjang tempoh pengajian saya. Sokongan daripada pihak fakulti sangat membantu dalam merealisasikan penyelidikan ini.

Tidak lupa kepada ibu bapa tercinta serta ahli keluarga yang sentiasa memberi kasih sayang, doa, dan semangat yang tidak pernah putus. Juga kepada rakan-rakan seperjuangan yang telah bersama-sama membantu, menyokong, dan berkongsi ilmu sepanjang tempoh pengajian ini. Tanpa bantuan dan sokongan kalian, projek ini tidak akan dapat disiapkan dengan jayanya.

Sekian, terima kasih.

RUJUKAN

- Ahmad, S., & Abdul Latif, N. F. (2023). Pengaruh tekanan akademik, keyakinan tingkah laku akademik dan kebimbangan dalam kalangan pelajar universiti. *EDUCATUM Journal of Social Sciences*, 9(2), 13–22. Retrieved from: <https://doi.org/10.37134/ejoss.vol9.2.2.2023>
- Ajay Bandi, P. H. (2014). *Usability Testing: A Software Engineering Perspective*. Maryville, MO: Northwest Missouri State University.
- Atlassian. (2024). *AI for project management: Tools and best practices*. Retrieved from: <https://www.atlassian.com/work-management/project-management/ai-project-management>
- Bhatla, S. (2024, October 28). Incorporating AI into mobile apps: Trends, challenges, and best practices. *Neuronimbus*. Retrieved from: <https://www.neuronimbus.com/blog/incorporating-ai-into-mobile-apps-trends-challenges-and-best-practices/>
- Chouinard, J. (2024, February 16). Unlocking enhanced productivity with AI on mobile devices: A comprehensive guide. *Varsity Technologies*. Retrieved from: <https://www.varsitytech.com/ai-on-mobile-devices/>
- Firebase. (n.d.). *Get started with Cloud Firestore*. Retrieved from: <https://firebase.google.com/docs/firestore/quickstart>
- Gala, A. S. (2024, November 28). User Acceptance Testing (UAT): Checklist, Types and Examples. *TestRail*. Retrieved from: <https://www.testrail.com/blog/user-acceptance-testing/>
- Google. (n.d.). *WorkManager overview*. Retrieved from: <https://developer.android.com/topic/libraries/architecture/workmanager>
- Jetpack Compose. (n.d.). *Android Developers Documentation*. Retrieved from: <https://developer.android.com/jetpack/compose>

- Kumar, A. (2023, May 31). Iterative vs. incremental development explained. *Built In*. Retrieved from: <https://builtin.com/software-engineering-perspectives/iterative-vs-incremental>
- Li, Y., et al. (2022). An empirical study of AI techniques in mobile applications. *arXiv.org*. Retrieved from: <https://arxiv.org/abs/2212.01635>
- Nepal, S., et al. (2024). From user surveys to telemetry-driven agents: Exploring the potential of personalized productivity solutions. *arXiv.org*. Retrieved from: <https://arxiv.org/abs/2401.08960>
- OpenRouter. (n.d.). *OpenRouter API Reference*. Retrieved from: <https://openrouter.ai/docs>
- Point, T. (n.d.). *JUnit – Writing a Test*. Tutorialspoint. Retrieved from: https://www.tutorialspoint.com/junit/junit_writing_tests.htm
- Ramli, N. H., & Sheikh Dawood, S. R. (2020). The relationship between study anxiety and academic performance among engineering students. *e-Bangi: Journal of Social Sciences and Humanities*, 17(5), 34-45. Retrieved from: <https://ejournal.ukm.my/ebangi/article/download/41607/10819>
- Stack Overflow. (n.d.). How to use WorkManager for periodic tasks in Android? Retrieved from: <https://stackoverflow.com/questions/54722656>
- World Economic Forum. (2019). *The Fourth Industrial Revolution: What it means, how to respond*. Retrieved from: <https://www.weforum.org/agenda/2019/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond>

Ahmad Hafiy Bin Ahmad Fizri (A200641)

Prof. Madya Dr. Abdul Hadi Bin Abd Rahman

Fakulti Teknologi & Sains Maklumat

Universiti Kebangsaan Malaysia