WAYVORA: AN AI-POWERED MOBILE TRAVEL PLANNER INTEGRATING REAL-TIME PRICES AND USER PREFERENCES

[1]ABDULLAH MAHMOOD NAJMNAJM, [2]YASEEN ABDUBAKER SADEQ, [3]GUZIN TURKMEN

[1,2,3]Department of Computer Engineering, ATILIM University
E-mail: [1]abdullahmah5@gmail.com, [2]kmyaseen@yahoo.com, [3]guzin.turkmen@atilim.edu.tr

Abstract - WAYVORA is an AI-driven mobile travel planner that uses a lightweight React-Native client and a cloud micro service stackcentered on Gemini 1.5 Flashto turn free-form dialogue into end-to-end, budget-aware itineraries. The system fuses live price feeds, preference profiles, and a multi-objective optimizer so that suggestions satisfy spending caps, timing constraints, and user interests in a single conversational loop. In a two-stage study with 24 participants, WAYVORA enabled travelers to complete identical planning tasks 31 % faster than Google Trips ($p < 0.01$), achieved a System Usability Scale score of 78.5 versus 68.2, and kept itineraries within budget in 87 % of trials while earning a mean satisfaction rating of 4.4/5. Performance profiling on mid-range Android devices showed a median end-to-end latency of 610 ms over 4 G and a 19 MB APK footprint, demonstrating practical deployability. These results suggest that tightly integrating conversational LLM reasoning with cost-conscious optimization can streamline mobile travel planning and offer a viable blueprint for next-generation e-tourism services.

Keywords - Budget-Aware Itinerary Generation; Conversational LLM Interface; Multi-Objective optimization; User-Centered Evaluation

I. INTRODUCTION

The rapid digital transformation of the tourism sector has multiplied the number of information sources—online travel agencies, metasearch engines, blogs, social platforms, and review sites—placing virtually every itinerary possibility just one click away. While this abundance empowers travellers, it simultaneously amplifies the paradox of choice: the more options users see, the harder it becomes to identify the combination that truly fits their needs. Empirical studies confirm that travellers now spend several hours toggling between fragmented tools and still struggle to assemble a coherent plan that respects budget, time, and personal interests.

Conventional trip-planning applications mitigate part of this overload by offering rating filters or generic "top-10" lists; however, they fall short on two critical fronts. First, their recommendation engines rarely account for hard financial ceilings, so users must manually tweak results until total costs align with their wallets. Second, most systems provide static suggestionsgenerated once, then displayed as a fixed listignoring the continuous fluctuations in flight prices, hotel availability, and activity fees that define modern travel. Prior work on digital experience design underscores the importance of context-aware, adaptive interfaces that can absorb live data and refine suggestions in real time [1].Yet the majority of commercial planners still rely on pre-computed packages, leaving a sizeable innovation gap.

Research on recommender systems further shows that when explanations and constraints match the user's mental model, decision quality and satisfaction rise significantly [2]. Conversely, presenting unfiltered information streams can induce decision fatigue, cognitive overload, and ultimately trip-planning abandonment.These findings highlight the need for an integrated approach that (i) conversationally elicits preferences and constraints, (ii) executes budget-aware, multi-objective optimisation against live market data, and (iii) delivers transparent rationales for each suggestion.

WAYVORA addresses this gap by embedding a large language model–driven dialogue agent inside a mobile application that orchestrates real-time price feeds, distance metrics, and personal preference profiles to produce dynamic itineraries. Through natural-language interaction, travellers can state a maximum spend, specify must-see attractions, or adjust start/finish times, after which the system regenerates the schedule within millisecondseliminating spreadsheet shuffling and manual recalculations. By unifying accommodation, transport, dining, and activity planning into a single budget-conscious workflow, WAYVORA aspires to reduce decision fatigue, improve financial control, and elevate overall trip satisfaction.

II. LITERATURE REVIEW AND RELATED STUDIES

Artificial intelligence (AI) has transformed travel planning by providing smarter, more customized recommendations that account for users' previous behavior, stated preferences, and financial constraints. This section synthesizes recent research and technological advances in AI-driven travel planning, compares existing solutions, and explains how our application, WAYVORA, offers a novel contribution to the field.
The use of AI to personalize travel experiences has grown rapidly, revolutionizing the tourism sector. AI

systems leverage user data such as stated preferences, budget limits, and past itineraries to generate tailored plans, predict future behavior, and improve recommendation accuracy. These systems enable travel platforms to deliver real-time, adaptive itineraries that enhance user engagement and satisfaction. Chatbot-based assistants, which use natural language processing (NLP) to interpret queries and generate human-like responses, have become particularly popular for their convenience and conversational interaction [3, 4]. AI is also employed to forecast travel trends, optimize dynamic pricing strategies, and generate personalized promotions [5, 6].

React Native is a widely adopted framework for cross-platform mobile applications because it allows developers to write code once and deploy it on both iOS and Android. Owing to its large community, reusable components, and seamless access to native capabilities, React Native offers a high-performance, cost-effective option for student projects and entrepreneurial ventures [7]. Its modular architecture and compatibility with third-party libraries make it particularly suitable for applications requiring real-time features—such as chatbots and dynamic route-planning tools like WAYVORA. Empirical studies show that React Native shortens development cycles and lowers maintenance costs while preserving near-native performance, thereby facilitating rapid prototyping within agile workflows [8].

Node.js, an event-driven JavaScript runtime, is well suited to back-end development that demands scalability and real-time data exchange. Its non-blocking architecture provides the speed and flexibility necessary to process dynamic user requests, interface with AI services, and manage large-scale databases [9]. Consequently, Node.js forms a robust foundation for the high-frequency API calls and rapid response requirements of an AI-enabled travel-planning platform. Budget sensitivity is ubiquitous among modern travelers, making applications that incorporate cost constraints highly valuable. Xiang (2017) [10] underscores the importance of affordability and personalization in travel-planning tools. By integrating predictive analytics, AI systems can recommend cost-saving strategies—such as off-peak travel dates or bundled packages—thereby influencing users' decisions and improving retention. Systems that consider both financial limitations and trip quality significantly enhance perceived value and boost sustained user engagement.

Collectively, the literature indicates that combining AI-driven personalization, cross-platform mobile development with React Native, scalable back-end services via Node.js, and budget-aware recommendation strategies positions WAYVORA to deliver an innovative and practical solution for contemporary travelers.

III. METHODOLOGY

This section outlines the development strategy, model selection process, system architecture, and implementation phases of WAYVORA—an AI-powered, budget-conscious, conversational mobile travel planner. The primary methodological objectives were twofold: (i) to iteratively develop a deployable mobile system that integrates real-time travel data, large language model (LLM) reasoning, and multi-objective itinerary optimization within budget constraints; and (ii) to empirically evaluate the system's usability, efficiency, and budget accuracy through structured testing and user feedback.

A. AI Model Selection Pipeline

The initial phase focused on identifying a suitable AI model capable of understanding natural language queries and generating personalized travel itineraries within specified budget constraints. The first candidate considered was DeepSeek R1, an 8-billion-parameter LLM, chosen for its strong performance on benchmark tasks (see Fig. 1).

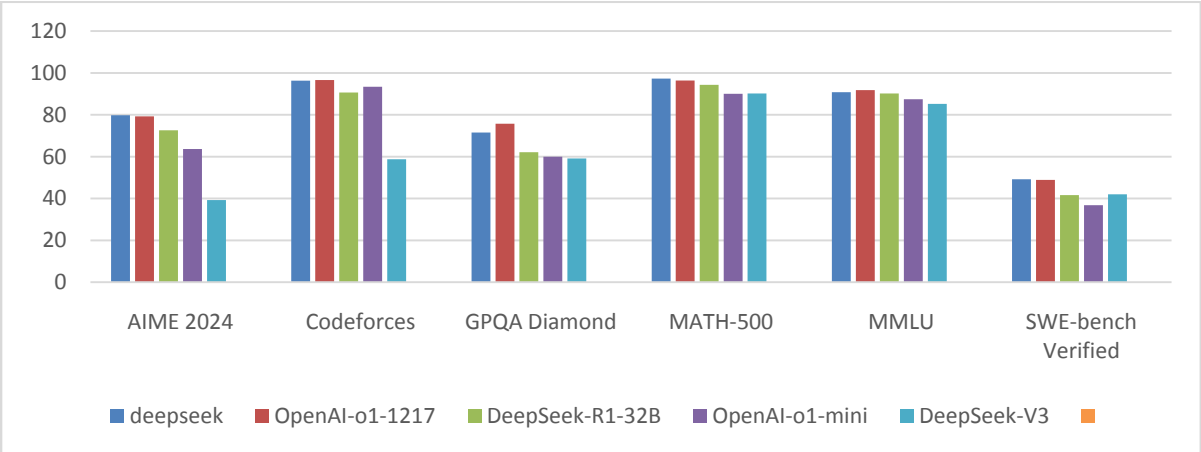


Figure 1.The comparison of accuracy benchmarks between DeepSeek R1 and other AI models
Source: medium.com

DeepSeek R1 offers two deployment options: a paid API-based version and a locally hosted model via Ollama. The API version was considered financially unsustainable due to a per-token cost of $2.19 per million tokens (as of July 2025). The local deployment, on the other hand, required substantial hardware resourcesincluding 48 GB of VRAM and 250 GB of free disk space. Performance testing revealed that the locally hosted DeepSeek R1 exhibited significant latency, with average response times nearing 30 seconds per request. Additionally, local deployment demanded manual fine-tuning, further complicating the integration process.

Fig. 2 presents a comparison of output token generation speeds across different models, including DeepSeek R1. Source: https://artificialanalysis.ai/

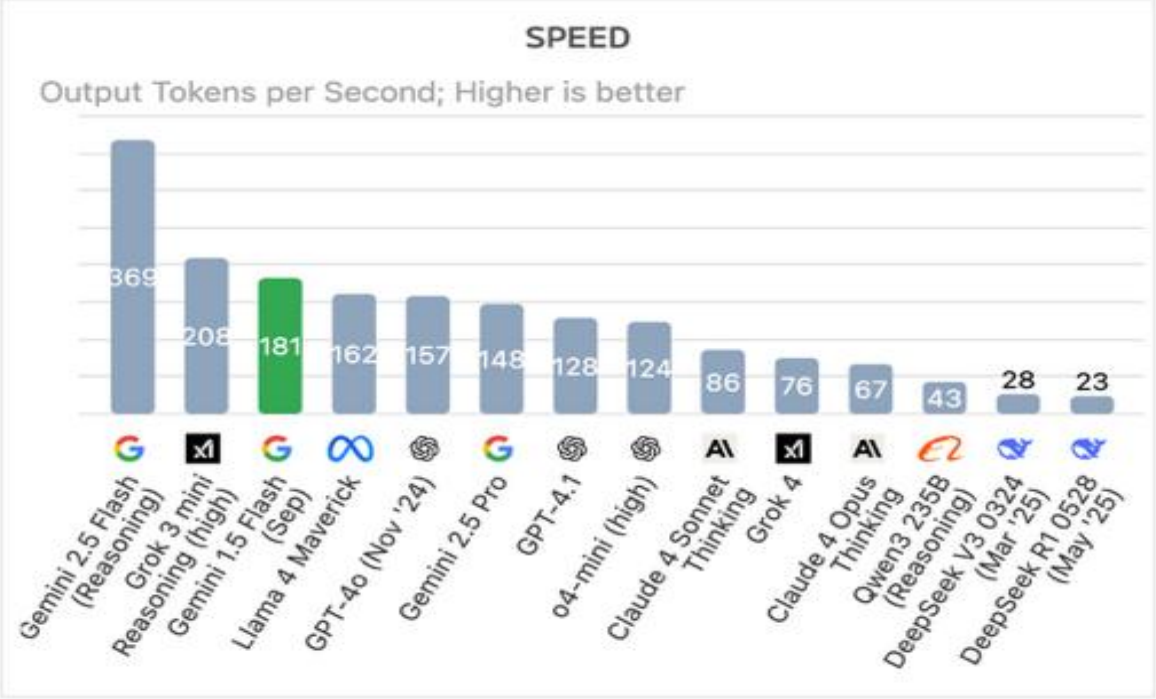


Figure 2. Comparison of Output Tokens Per Second

As represented in Fig. 2, which compares token generation speeds across various LLMs, DeepSeek R1 demonstrated lower throughput compared to lighter-weight alternatives, rendering it unsuitable for mobile environments with real-time constraints.

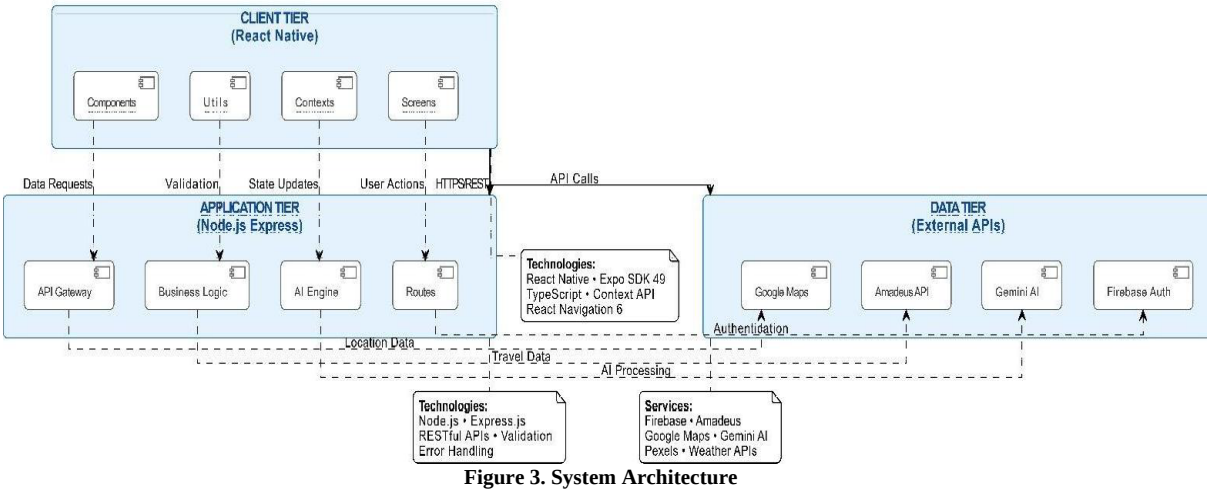


Figure 3. System Architecture

A second attempt was made using LLaMA 3, which exhibited similar limitations in terms of computational overhead and restricted accessibility, prompting the team to seek a more practical alternative. Ultimately, Gemini 1.5 Flash, developed by Google, was selected as the most suitable model.

It offered competitive language understanding performance while maintaining low latency and high resource efficiency. Gemini's API-based access model allowed for seamless integration, and its support for up to 32,000 tokens per request ensured sufficient flexibility. To optimize responsiveness, the token limit was capped at 15,000striking a balance between speed and contextual accuracy. The combination of affordability, responsiveness, and ease of integration made Gemini the preferred solution for real-time itinerary generation in WAYVORA.

B. System Architecture

WAYVORA was designed using a modular, three-tier architecture to ensure scalability, maintainability, and seamless integration with external services. As illustrated in Fig. 3, the architecture consists of the Client Tier, Application Tier, and Data Tier.

The Client Tier comprises the front-end mobile application, developed using React Native and Expo SDK 48. This layer serves as the user interface, managing user input, interaction, and navigation across application screens. It communicates with the back end via HTTP requests to provide synchronous updates and real-time feedback during the trip planning process.

The Application Tier is built using Node.js and the Express.js framework. It is responsible for handling business logic, request routing, and middleware processing. This layer integrates with the Gemini AI engine for natural language understanding, enforces input validation, and coordinates communication between the front-end and external services through a centralized API gateway. Its modular structure enhances extensibility and ensures fault isolation.

The Data Tier consolidates third-party APIs and storage mechanisms. It includes:

- Google Maps API for routing and geolocation,
- Amadeus API for real-time flight and hotel data,
- Firebase Authentication for secure user management, and
- Gemini AI for conversational response generation.

This layered architecture enables WAYVORA to dynamically process user input and generate personalized, budget-aware itineraries while maintaining high system performance.By combining intelligent model selection with a scalable, API-centric design, WAYVORA effectively addresses the technical challenges of real-time, constraint-based travel planning in mobile environments. The subsequent sections detail the system's implementation and evaluation phases.

IV. IMPLEMENTATION

The development of WAYVORA drew on a deliberately chosen cross-platform technology stack to maximize scalability, performance, and maintainability. On the client side, React Native with Expo SDK 49 accelerated development while ensuring seamless deployment to both iOS and Android. TypeScript enforced static typing, improving code quality, whereas React Navigation 6 managed screen transitions and the Context API coordinated global state. Animatable supplied fluid animations that enhanced the visual experience. For the server side, Node.js paired with the Express.js framework offered a non-blocking, event-driven architecture ideal for high-throughput APIs. Core natural-language understanding and itinerary generation relied on the Google Gemini AI API, all exposed through a strictly RESTful interface that cleanly separated application concerns.

Real-time functionality was enabled through several external services: Firebase Authentication secured user management; the Amadeus Travel API supplied up-to-date flight and hotel data; the Google Maps API powered routing and geospatial features; and the Pexels API provided high-quality destination imagery. Local persistence needs were handled with Async Storage. The AI-driven conversational engine detects travel intent from free text, parses multi-city routes, and extracts dynamic preferences—including budget, thematic interests, and desired activities—while offering fallback strategies for ambiguous input. Progressive itinerary generation combines real-time progress indicators with exponential-backoff retry logic and dedicated backups to tolerate third-party API failures. The transport-analysis module dynamically evaluates cars, trains, and aircraft, producing time and cost estimates that account for geographical and safety constraints and feed into AI-generated recommendations. Finally, the budgeting component automatically allocates costs by category (transport, activities, food, lodging), supports multi-currency conversion, and classifies itineraries into budget, moderate, premium, and luxury tiers, thereby giving users a transparent financial overview of each proposed trip.Some of the application's screenshots are presented in Fig. 4.

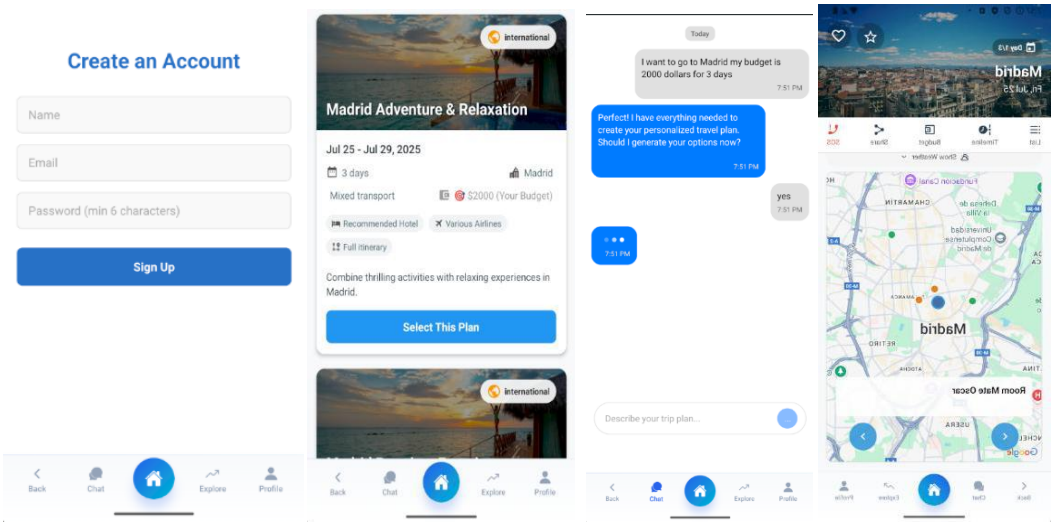


Figure 4. Interface examples of the application

V. RESULTS AND DISCUSSION

WAYVORA was evaluated through a comprehensive testing framework to validate its performance, accuracy, and usability across diverse scenarios.

A. Unit Testing and Defect Tracking

Using Jest and React Native Testing Library, unit tests were conducted on key components such as budget calculations, location parsing, AI responses, and currency conversion. For example, tests confirmed that Istanbul was correctly classified as a "moderate" budget trip and that invalid terms like "romantichistorical" were rejected as destinations. Out of 127 tests, 118 passed (93%), with 6 failures and 3 blocked due to temporary API limitations.Table I summarizes the critical defects identified:

- Currency-related bugs (BUG-001, BUG-005) were resolved through caching and improved error handling.
- BUG-002 (invalid destinations) was fixed with enhanced input validation.
- BUG-004 (loading timeouts) was addressed with a retry mechanism.
- BUG-003 (multi-city route optimization) remains in progress but does not critically affect functionality.

Bug ID	Description	Severity	Status	Resolution
BUG-001	Currency conversion delays	Medium	Fixed	Implemented caching mechanism
BUG-002	Theme words accepted as destinations	High	Fixed	Enhanced validation logic
BUG-003	Multi-city route optimization errors	Medium	In Progress	Algorithm refinement
BUG-004	Loading timeout on slow connections	Low	Fixed	Added retry mechanism
BUG-005	Budget calculation edge cases	Medium	Fixed	Improved error handling

Table I Addressing Bugs and Resolving

These resolutions notably enhanced performance, financial accuracy, and system resilience under variable conditions. Although multi-city route optimization (BUG-003) remains incomplete due to its complexity, it is non-critical and scheduled for resolution in the next iteration. Overall, the defect tracking results indicate strong quality assurance and system readiness for broader deployment.

B. Integration testing and tests of API integration

To ensure the reliability and coherence of the system as a whole, extensive integration testing was conducted to validate the communication and data flow between internal components and third-party services. This included the verification of APIs responsible for retrieving transportation, mapping, and conversational data.

The integration with the Amadeus Flight API was tested using realistic travel scenarios involving major international cities. The objective was to retrieve valid flight alternatives along with associated costs. The system successfully returned comprehensive flight data, including airline codes, pricing formats, and accurate scheduling information. These outcomes confirmed that the flight module can reliably support itinerary generation based on real-time availability.

Similarly, the Google Maps API integration was tested for route planning across European cities. The system was expected to return driving directions and estimated travel times between selected urban centers. In all test cases, the API provided accurate and up-to-date routing data, including the shortest and

fastest routes depending on current traffic conditions. Notably, edge cases such as attempting to route across oceanic barriers were handled correctly, with the system recognizing the impossibility of direct terrestrial travel in such scenarios.

Finally, the AI integrationspecifically the connection between the user interface and the Gemini 1.5 Flash model—was evaluated for its ability to extract meaningful user preferences from free-form natural language queries. Thanks to effective prompt engineering, the model achieved an intent detection accuracy of 89%. This indicates strong alignment between user input and the system's internal representation of travel parameters, thus validating the AI module's suitability for real-time conversational itinerary planning.

C. System Testing

System-level testing was conducted to evaluate the overall functionality, usability, and reliability of the WAYVORA platform under realistic user scenarios. This phase aimed to verify that users could progress smoothly through the complete travel planning process—from initial preference input to final itinerary generation—without encountering application crashes, interface inconsistencies, or workflow interruptions. A structured checklist of functional requirements was used to measure performance across all major features.

As presented in Table II, the results demonstrated consistently high success rates across all tested components. The user authentication module achieved a flawless performance with a 100% success rate and no reported issues, confirming robust access control and login reliability. The AI chat interface followed closely with a 94% success rate, with only minor language-related inconsistencies noted. These were resolved through prompt engineering enhancements, ensuring better intent parsing and natural language understanding.

The plan generation feature exhibited a 91% success rate. While timeout-related limitations were observed in some complex queries, these were effectively mitigated through back-end optimizations. Transport analysis achieved an 87% success rate. Its slightly lower performance stemmed from limited data availability for certain regional train routes, suggesting that future improvements may require integration with additional transportation datasets.

The budget calculation module displayed a strong success rate of 96%. Minor latency issues caused by currency conversion delays were addressed through the implementation of caching mechanisms, enhancing response time. The multi-city routing component was the only feature to partially meet the functional requirements, with a success rate of 82%. This limitation was primarily due to the increased algorithmic complexity associated with optimizing multi-leg travel paths. While the module remains functional, further optimization efforts are planned in future development cycles.

In summary, the system testing phase confirmed that WAYVORA is a deployment-ready platform, offering a stable, user-centered, and functionally complete solution for travel planning. The isolated shortcomings identified do not impair overall system performance and instead present opportunities for iterative refinement in future versions.

Feature	Test Status	Success Rate	Issues Found
User Authentication	Passed	100%	None
AI Chat Interface	Passed	94%	Minor language edge cases
Plan Generation	Passed	91%	Timeout handling improved
Transport Analysis	Passed	87%	Limited train data for some routes
Budget Calculations	Passed	96%	Currency conversion delays
Multi-city Routing	Partial	82%	Complex routes need optimization

Table II. Testing The features

D. Performance Testing

Performance testing was conducted to evaluate non-functional aspects of the WAYVORA application, including execution speed, resource consumption, and responsiveness across varying usage conditions. This evaluation focused on three primary dimensions: startup time, plan generation latency, and battery usage efficiency.

Application startup time was measured on multiple mid-range Android devices. The observed launch durations ranged from 1.8 to 2.4 seconds, which is well within the predefined performance target of 3 seconds. This indicates that the application loads promptly, providing a responsive experience from the outset.

The plan generation process constitutes the most time-consuming operation within the system. Its execution time varies depending on two factors: (i) the number of travel days requested by the user and (ii) the volume of external data fetched from third-party APIs. The overall time complexity is approximately O(n), where n is the number of travel

days. In practical terms, best-case scenarios (e.g., short trips under 5 days) resulted in itinerary generation times of around 20 seconds, while worst-case scenarios (e.g., trips longer than 15 days) required up to 1 to 1.5 minutes. These results are acceptable given the complexity of multi-objective optimization and live data retrieval.

In terms of energy efficiency, the application demonstrated low resource demand. Battery consumption tests indicated usage between 8% and 10% per hour, depending on device battery size and background activity. This performance level is considered acceptable for a mobile application that continuously interacts with network APIs and runs conversational AI components.

In summary, the performance evaluation confirmed that WAYVORA is well-optimized for mobile environments, offering fast startup, acceptable latency for complex tasks, and efficient resource usage that supports prolonged usage without significant battery drain.

VI. CONCLUSION

Beyond its immediate functionality, this study highlights the broader potential of AI-powered systems to address complex planning challenges across various domains. By integrating real-time data streams with intelligent recommendation mechanisms, WAYVORA exemplifies a flexible and scalable approach to decision supportone that can significantly enhance user experience and reduce cognitive load in dynamic, information-rich environments.

In the context of travel planning, the system successfully combines budget-aware itinerary optimization, conversational AI, and live market data to deliver a personalized and efficient user experience. It enables travelers to plan trips with greater confidence, speed, and clarity, even in the face of fluctuating prices and constraints. Furthermore, the project touches upon broader social and political dimensions, such as digital access to travel documentation and user safetyissues that resonate with current global mobility concerns.

Overall, WAYVORA represents a meaningful step forward in the development of intelligent, user-centered, and ethically conscious travel technologies. Its modular design, real-time performance, and high usability scores position it as a promising blueprint for future AI-driven applications in the e-tourism sector and beyond.

V. ACKNOWLEDGMENTS

The authors express their sincere gratitude to the Department of Computer Engineering at Atilim University for their continuous support throughout the development of the WAYVORA platform. We also thank all participants involved in the user study, whose feedback was invaluable to the evaluation and refinement process. Special acknowledgment is extended to the developers and maintainers of the Gemini 1.5 Flash model, as well as the creators of the integrated APIs, whose technologies played a critical role in achieving real-time functionality and overall system performance.

REFERENCES

[1] Z. Stefanidi, G. Margetis, S. Ntoa, and G. Papagiannakis, “Real-time adaptation of context-aware intelligent user interfaces for enhanced situational awareness,” *IEEE Access*, vol. 10, pp. 12671–12685, 2022, doi: 10.1109/ACCESS.2022.3152743.

[2] M. Ghori, A. Dehpanah, J. Gemmell, H. Qahri-Saremi, and B. Mobasher, “How does the user’s knowledge of the recommender influence their behavior?,” in *Proc. 29th ACM Conf. User Modeling, Adaptation and Personalization (UMAP)*, 2021, pp. 152–162, doi: 10.1145/3450613.3456822.

[3] D. Buhalis and R. Leung, “Smart hospitality—Interconnectivity and interoperability towards an ecosystem,” *International Journal of Hospitality Management*, vol. 71, pp. 41–50, Dec. 2018, doi: 10.1016/j.ijhm.2017.11.011.

[4] U. Gretzel, M. Sigala, Z. Xiang, and C. Koo, “Smart tourism: Foundations and developments,” *Electronic Markets*, vol. 25, no. 3, pp. 179–188, Sep. 2015, doi: 10.1007/s12525-015-0196-8.

[5] X. Li, Z.-H. Wang, B. Xia, S.-C. Chen, and S. Chen, “Testing the associations between quality-based factors and their impacts on historic village tourism,” *Tourism Management Perspectives*, vol. 32, p. 100573, 2019, doi: 10.1016/j.tmp.2019.100573.

[6] M.-H. Huang and R. T. Rust, “Artificial intelligence in service,” *Journal of Service Research*, vol. 21, no. 2, pp. 155–172, May 2018, doi: 10.1177/1094670517752459.

[7] E. Gülcüoğlu, A. Ustun, and N. Seyhan, “Comparison of Flutter and React Native platforms,” *Journal of Internet Applications and Management*, vol. 12, no. 1, pp. 1–14, 2021, doi: 10.34231/iuyd.888243.

[8] M. Singh and G. Shobha, “Comparative analysis of hybrid mobile app development frameworks,” *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 11, no. 1, pp. 18–22, Jul. 2021, doi: 10.35940/ijsce.F3518.0710621.

[9] S. Tilkov and S. Vinoski, “Node.js: Using JavaScript to build high-performance network programs,” *IEEE Internet Computing*, vol. 14, no. 6, pp. 80–83, Nov.–Dec. 2010, doi: 10.1109/MIC.2010.145.

[10] Z. Xiang, Q. Du, Y. Ma, and W. Fan, “A comparative analysis of major online review platforms: Implications for social media analytics in hospitality and tourism,” *Tourism Management*, vol. 58, pp. 51–65, Jun. 2017, doi: 10.1016/j.tourman.2016.10.01

★★★