© March 2026 | IJIRT | Volume 12 Issue 10 | ISSN: 2349-6002

Heymate : One Tap, Any Task, Any Time

Harinni G[1], Sruthi B[2], Dr. Surya Dharshini V[3], M Victor Jose[4]
[1,2,3,4]*Department of Computer Science and Engineering*
[1,2,3,4]*Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College Chennai, India*

***Abstract*—Contemporary on-demand digital service platforms predominantly operate within isolated operational domains, such as transportation, food delivery, household assistance, and emergency response. Although these systems leverage real-time geolocation and cloud-based coordination, their domain-specific architecture results in fragmented service ecosystems, limited cross-functional integration, and inefficient resource utilization issues. This study presents HEYMATE, a unified multidomain hyperlocal service orchestration framework designed to consolidate heterogeneous urban services within a single scalable architecture.**

The system identifies nearby service providers based on the user's location and ranks them by considering their past performance and user feedback. For regular service requests, providers can submit quotations within a specified timeframe, allowing users to choose the most suitable option. The platform also allows individuals to switch between being service requesters and providers, making participation more flexible. In urgent situations, the system prioritizes requests and directly assigns the nearest available provider to reduce the response time.

The overall architecture follows a client-server model in which mobile applications communicate with a centralized server. The server handles location-based filtering, stores user and service data, and tracks the status of each request from its initiation to its completion. This structured approach helps manage resources efficiently and supports coordinated service delivery across several categories.

***Index Terms*—Multi-Domain Orchestration, Hyperlocal Allocation, Dynamic Role Scheduling, Bid-Based Task Assignment, Emergency Response Routing**

I. INTRODUCTION

The rapid growth of mobile computing, cloud infrastructure, and geospatial technologies has significantly transformed urban services accessibility and delivery. On-demand digital platforms have emerged as critical intermediaries that connect consumers with service providers in real time. These platforms typically rely on geolocation tracking, centralized databases, and cloud-based orchestration engines to facilitate service requests, provider matching, and the completion of transactions. Although such systems have enhanced operational efficiency within specific domains, they remain fundamentally limited by domain-centric architectural designs.

Most contemporary platforms are vertically specialized, meaning that they are engineered to handle a single category of services. Transportation applications focus exclusively on ride-hailing, food delivery platforms that manage restaurant-to-consumer logistics, and household service aggregators that coordinate professional assistance bookings. This vertical isolation compels users to interact with multiple independent applications to fulfil different service needs, thereby increasing cognitive load and operational redundancy. From a systems perspective, such fragmentation restricts unified coordination, prevents cross-domain resource sharing, and limits adaptive load balancing across heterogeneous services.

Many existing platforms assign providers using fixed-matching rules or predefined pricing models. These systems usually do not allow competitive bidding, flexible price negotiations, or intelligent prioritization based on urgency. In critical situations, there is often no separate mechanism to fast-track emergency requests because they are processed through the same standard workflow as regular tasks. Moreover, most platforms clearly separate users into consumers and providers without allowing them to easily switch roles within the same system.

To overcome these challenges, this study introduces HEYMATE, a unified hyperlocal service platform that supports multiple service categories within a single framework. The system processes different types of requests through a common architecture and uses location-based filtering, quotation-based selection, emergency handling, and performance-based ranking to manage the allocations. It also enables users to act as both service requesters and providers. The goal is to create a flexible and efficient coordination system that can adapt to varying service demands in urban areas.

II. RELATED WORK

Many domain-specific platforms have improved the delivery of services in real time. Ride-hailing applications use location tracking to quickly match drivers with passengers and reduce waiting times. Food delivery platforms combine restaurant listings, order management systems, and delivery tracking to ensure faster processing. Quick-commerce services focus on rapid dispatch from nearby warehouses, whereas home service platforms use scheduled booking systems to manage maintenance and repair requests efficiently.

Although these platforms perform well within their own areas, they are built to handle only one type of service. Each application operates with its own database, provider network, and allocation process without interacting with other service categories. Consequently, there is little coordination across domains. Most research on hyperlocal services also concentrates on improving distance-based matching or balancing workload within a single category. While some systems use ratings or bidding in limited ways, these features are rarely combined into a single framework that can manage different types of services under a common structure.

In addition, users are usually restricted to a single role, either as a service requester or provider. This rigid separation limits flexibility and reduces opportunities for broader participation in the field. Emergency handling, when available, is generally designed only for a specific service and does not function as part of a larger coordination system. These limitations highlight the need for a unified framework that can process different types of service requests, allow users to switch roles when required, support competitive bidding, and efficiently handle urgent tasks within a single integrated platform.

III. PROBLEM FORMULATION

Currently, urban service platforms primarily function as separate, domain-focused applications. Transportation, food delivery, home services, and emergency assistance are handled by different apps that do not interact with one another. Consequently, users must switch between multiple platforms to meet their daily and urgent needs. This separation increases the effort required by users and leads to fragmented service coordination. Because each platform maintains its own provider network and operational logic, there is no shared optimization of resources across domains. From a system design perspective, this isolation limits the overall efficiency, prevents balanced workload distribution, and restricts consistent quality-of-service management across different service categories.

Most on-demand platforms use allocation strategies specifically designed for a single type of service. For example, ride-hailing systems focus on assigning the nearest driver, whereas food delivery applications prioritize orders based on queue position and delivery time. Although these methods are effective within their respective domains, they are closely tied to domain-specific parameters. They are not built with a generalized structure that can handle different types of services under one framework. Therefore, extending such systems to support multi-domain coordination would require major architectural changes. In many cases, resource allocation remains reactive and relies on basic matching rules rather than adaptive or priority-aware mechanisms.

Another important limitation is the rigid separation between consumers and the providers. Most platforms assign users a fixed role, either as someone who requests services or provides them. However, in real-world situations, individuals may switch roles depending on their availability, skills or needs. For example, a person who requests a service today might be capable of providing a different service the next day. When platforms do not support this flexibility, overall participation and service availability are reduced. A more adaptable framework should allow users to transition between roles without requiring separate accounts or system reconfigurations.

Emergency handling is also limited in several existing systems. Some platforms offer faster services through surge pricing or premium options; however, these are usually based on commercial factors rather than urgency. There is often no structured mechanism to identify and prioritize critical requests. Consequently, urgent tasks may be processed in the same way as routine tasks, which can delay response times during serious situations. A more effective system should include a clear priority mechanism that adjusts allocation decisions based on the urgency, location, and service type.

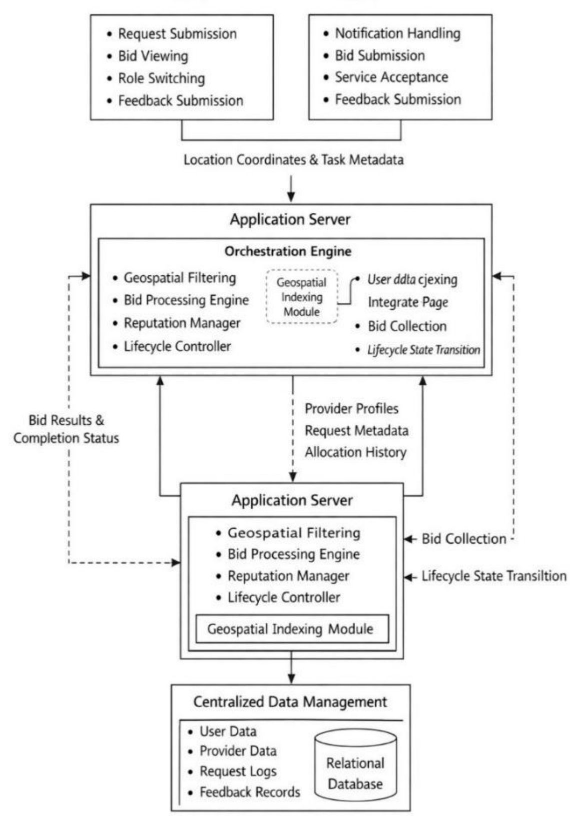
Collectively, these limitations highlight the need for a unified and scalable coordination framework that can operate across multiple service domains. Such a system should be able to process different types of requests using a common structure, allocate providers based on location and availability, allow flexible role transitions, and prioritize urgent tasks. By separating service logic from domain-specific constraints and introducing intelligent coordination mechanisms, a unified urban service platform can improve interoperability, scalability, and resource utilization in modern cities.

IV. PROPOSED UNIFIED FRAMEWORK

HEYMATE is designed with a common service structure so that different types of requests can be handled in the same way. Every request includes basic details such as the service category, user location (latitude and longitude), priority level, time of request, and other task-related information. By following this uniform format, the system can process transportation, delivery, household, or emergency services using the same core workflow without changing the underlying architecture.

The platform also allows users to act either as service requesters, service providers, or both. Instead of fixing a user's role permanently, the system permits role switching based on login credentials and access permissions. A person can request a service at one time and provide a different service at another time without creating multiple accounts. This flexibility increases participation and helps improve overall service availability.

To identify suitable providers, the system first applies location-based filtering. Only providers within a defined service radius are considered for a request. The allocation process then depends on the priority level of the task. For normal requests, eligible providers are invited to submit price quotations within a specified time limit. After the bidding period ends, the system compares the received quotations based on price, estimated arrival time, and provider rating. For emergency requests, the bidding stage is skipped, and the closest available provider is assigned immediately to reduce response time. After task completion, user feedback is recorded and the provider's rating is updated, ensuring that future allocations consider past performance.



V. SYSTEM ARCHITECTURE AND WORKFLOW

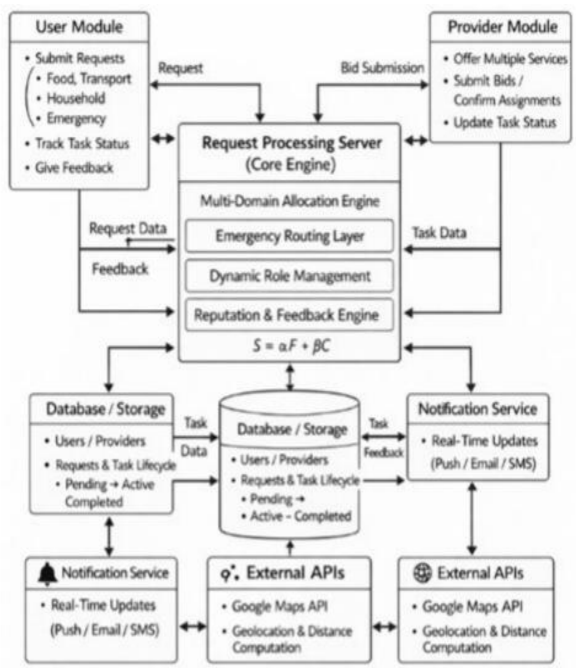
The HEYMATE platform follows a client-server architecture consisting of mobile applications for consumers and providers, an application server, and a centralized database. The mobile applications allow users to create requests, view bids, receive notifications, and submit feedback. The application server handles request validation, provider filtering, bid management, emergency routing, and status updates. The database stores user information, provider profiles, service categories, request details, bid records, and feedback data.

When a request is submitted, the server first verifies the selected category and priority level. It then performs location-based filtering using stored coordinates to identify nearby providers who match the requested service and are currently available. These providers form the candidate list.

For standard requests, a bidding window is opened during which providers can submit their quotations. Once the time limit expires, the system ranks the bids based on cost, estimated response time, and provider rating before selecting the most suitable option.

For emergency requests, the system immediately selects the nearest available provider without initiating bidding.

After assignment, the request moves through defined states such as pending, active, and completed. Once the task is finished, feedback is collected and the provider’s rating is updated. This step-by-step workflow ensures proper tracking, prevents duplication, and maintains consistent coordination across service categories.



VI. INTELLIGENT ALLOCATION AND BINDING ALGORITHM

The allocation process is handled entirely at the server level, where location processing and task control logic are combined to ensure accurate and conflict-free assignment. Whenever a user submits a request, the system first captures the geographical coordinates of the requester in terms of latitude and longitude. These coordinates form the basis for identifying nearby service providers. A radius-based search is then executed to retrieve providers who fall within a predefined service boundary and who are registered under the requested service category. This initial filtering step significantly reduces unnecessary comparisons and improves allocation speed.

Let

R denote the maximum service radius,

P_i denote the geographical coordinates of provider i,

U denote the geographical coordinates of the user.

The distance between U and P_i is computed using a geodesic distance formula that accounts for the curvature of the Earth. A provider is added to the candidate set C only if the following conditions are satisfied:

Distance $(U, P_i) \leq R$

Provider availability status = Active

Service category match = True

By enforcing these constraints, the system ensures that only relevant and available providers are considered for further processing.

For requests marked as normal priority, the system initiates a controlled bidding phase. Each provider in the candidate set C receives a structured task notification containing essential details such as service description, location, and expected time window. Providers are allowed to submit a quotation q_i within a predefined time interval T. All quotations are temporarily stored in a bid buffer until the bidding window closes. Once the interval T expires, the system evaluates the collected bids. The ranking may be based on minimum price, estimated arrival time, provider reputation score, or a weighted combination of these parameters. This competitive mechanism encourages fair pricing and allows users to benefit from transparent selection.

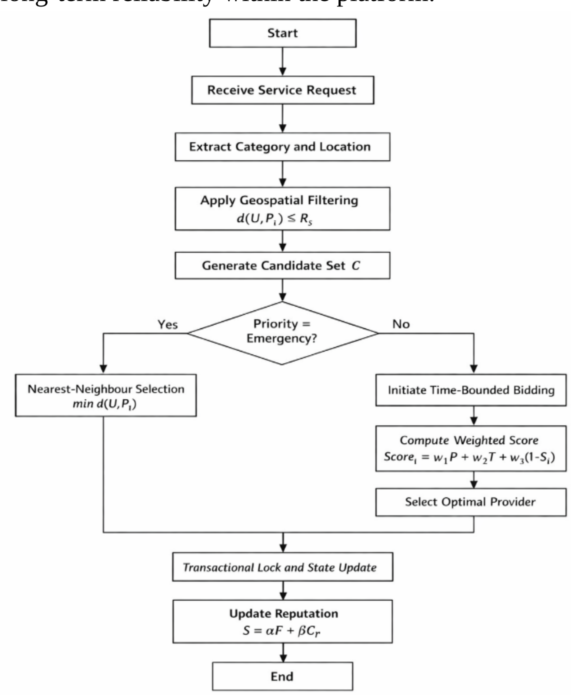
In contrast, for emergency-priority tasks, the bidding stage is bypassed entirely to avoid delay. In such cases, the algorithm performs a nearest-neighbor evaluation and selects the provider with the smallest computed distance from the user’s location. The assignment is triggered immediately, ensuring faster response in time-sensitive situations.

After a provider accepts the task, the system performs a transactional state update to lock the assignment. This prevents multiple providers from being assigned to the same request and avoids race conditions in concurrent environments. Upon completion of the task, the provider's performance metrics are updated. The reputation score S is recalculated using the expression:

$S = \alpha F + \beta C$

where F represents the normalized feedback rating provided by the user, and C represents the provider's consistency in completing assigned tasks successfully. The parameters α and β serve as weighting coefficients that determine the relative importance of feedback quality and completion reliability.

By continuously updating the reputation score, the system maintains a performance-based ranking mechanism. Providers who consistently deliver quality service and complete tasks reliably are more likely to be prioritized in future allocations. This adaptive scoring approach not only improves overall service quality but also promotes accountability and long-term reliability within the platform.



VII. IMPLEMENTATION AND PERFORMANCE DISCUSSION

The system is implemented using a layered structure that connects mobile front-end applications with a cloud-based backend server. Location-based filtering is supported through mapping services that provide coordinate extraction and radius queries. Role-based access control ensures secure authentication and enables users to switch roles when required. Request states are managed using structured status flags to maintain consistency and prevent duplicate assignments.

From a performance perspective, limiting provider search to a defined geographic radius reduces computation time and improves response speed. The bidding mechanism promotes competitive pricing and fair selection among providers. Emergency routing reduces delay by removing negotiation steps. The modular server design allows horizontal scaling through stateless APIs and load balancing, making the system capable of handling increased demand across different service types.

VIII. CONCLUSION

This study presented HEYMATE, a unified multi-domain hyperlocal service coordination framework designed to consolidate heterogeneous urban services within a single scalable architecture. The proposed model integrates proximity-based provider discovery, quotation-driven allocation, dynamic role management, and emergency priority routing to enable adaptive and efficient task orchestration.

By employing geospatial filtering, transactional state control, and reputation-weighted routing, the framework enhances allocation efficiency and structured lifecycle tracking. This architecture supports flexible participation and scalable deployment across diverse operational categories.

The proposed system establishes a foundation for integrated urban service management and enables future extensions, such as predictive demand modelling, automated pricing optimization, and intelligent load balancing.

ACKNOWLEDGMENT

The authors express their sincere gratitude to Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College for providing the institutional support, research infrastructure, and academic environment necessary for the successful completion of this work. The encouragement and guidance from the Department of Computer Science and Engineering significantly contributed to the conceptual development, system design, and technical refinement of the proposed HEYMATE framework.

The authors also acknowledge the valuable suggestions and constructive feedback offered by faculty members and peers during various stages of implementation and evaluation. Their insights into architectural modeling, intelligent allocation strategies, and performance considerations helped enhance the overall quality and rigor of the study.

REFERENCES

[1] Uber Technologies Inc., "Ride Hailing Service Platform," [Online]. Available: https://www.uber.com.

[2] Swiggy Pvt Ltd., "Online Food Delivery System," [Online]. Available: https://www.swiggy.com.

[3] Blinkit Commerce Pvt Ltd., "Quick Commerce Delivery Model," [Online]. Available: https://www.blinkit.com.

[4] Urban Company Services Pvt Ltd., "Home Service Aggregation Platform," [Online]. Available: https://www.urbancompany.com.

[5] Google Maps Platform, "Geospatial API Services for Location Processing," [Online]. Available: https://cloud.google.com/maps-platform.

[6] S. Sharma, R. Kumar, "Hyperlocal Service Allocation using Proximity Aware Algorithms," International Journal of Computer Applications, vol. 182, no. 12, pp. 15–22, 2021.

[7] M. Gupta, P. Singh, "Dynamic Role Management in On-Demand Service Platforms," IEEE Access, vol. 9, pp. 12345–12354, 2021.

[8] L. Chen, Y. Zhao, "Emergency Priority Routing in Urban Service Systems," Procedia Computer Science, vol. 183, pp. 345–352, 2021.

[9] J. Li, K. Wang, "Reputation Based Resource Allocation for Multi Domain Service Systems," Journal of Network and Computer Applications, vol. 175, pp. 102–111, 2021.

A. Singh, R. Verma, "Scalable Hyperlocal Task Management in Mobile Service Platforms," International Journal of Advanced Computer Science and Applications, vol. 12, no. 6, pp. 78–87, 2021.

[10] G. Adomavicius and A. Tuzhilin, "Context-Aware Recommender Systems," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 2, pp. 1–15, 2022.

[11] Y. Wang, D. Zhang, and L. Chen, "A Survey on Spatial Crowdsourcing: Algorithms and Applications," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 3, pp. 1–37, 2022.

[12] X. Tong, J. She, and L. Zhang, "Efficient Task Assignment in Spatial Crowdsourcing Under Budget Constraints," *IEEE Trans. Mob. Comput.*, vol. 21, no. 4, pp. 1503–1517, 2022.

[13] H. Chen and K. Wang, "Geospatial Indexing Techniques for Real-Time Location-Based Services," *Future Gener. Comput. Syst.*, vol. 134, pp. 92–104, 2023.

[14] M. Abdel-Basset, R. Mohamed, and V. Chang, "Hybrid Optimization for Smart Urban Service Allocation," *Sustain. Cities Soc.*, vol. 88, 2023.

[15] S. Kumar and R. Singh, "Hyperlocal Demand Prediction Using Machine Learning," *Appl. Soft Comput.*, vol. 137, 2023.

[16] H. Wu, X. Li, and Y. Chen, "Nearest Neighbor Search Optimization for Large-Scale Location Systems," *Inf. Sci.*, vol. 629, pp. 320–335, 2023.

[17] D. Zhang and Y. Wang, "Task Assignment Algorithms in Spatial Crowdsourcing: A Review," *ACM Comput. Surv.*, vol. 56, no. 2, 2023.

[18] E. Ahmed and M. Hossain, "Priority-Based Service Routing in Smart Cities," *IEEE Access*, vol. 11, pp. 78645–78658, 2023.

[19] K. Lin and S. Luo, "Performance Modeling of Multi-Domain Service Orchestration," *J. Parallel Distrib. Comput.*, vol. 176, pp. 112–124, 2023.

[20] R. Mehta, P. Sharma, and N. Gupta, "Reinforcement Learning for Dynamic Allocation in On-Demand Platforms," *Knowl.-Based Syst.*, vol. 266, 2024.