

**Personal Travel Assistant (PTA)**

# **1. Introduction**

The Personal Travel Assistant (PTA) is a backend solution designed to simplify travel planning. It is built with FastAPI (for REST APIs) and MongoDB (for storage).

PTA integrates:

* 🌦 Weather API → Provides live weather forecasts with travel safety suggestions.
* 🚌 Bus Listings → APSRTC, TSRTC, and Private travel buses.
* 🏨 Hotels API → Hotels with ratings & pricing.
* 💾 MongoDB Atlas → Stores user bookings securely.

# **2. Use Case**

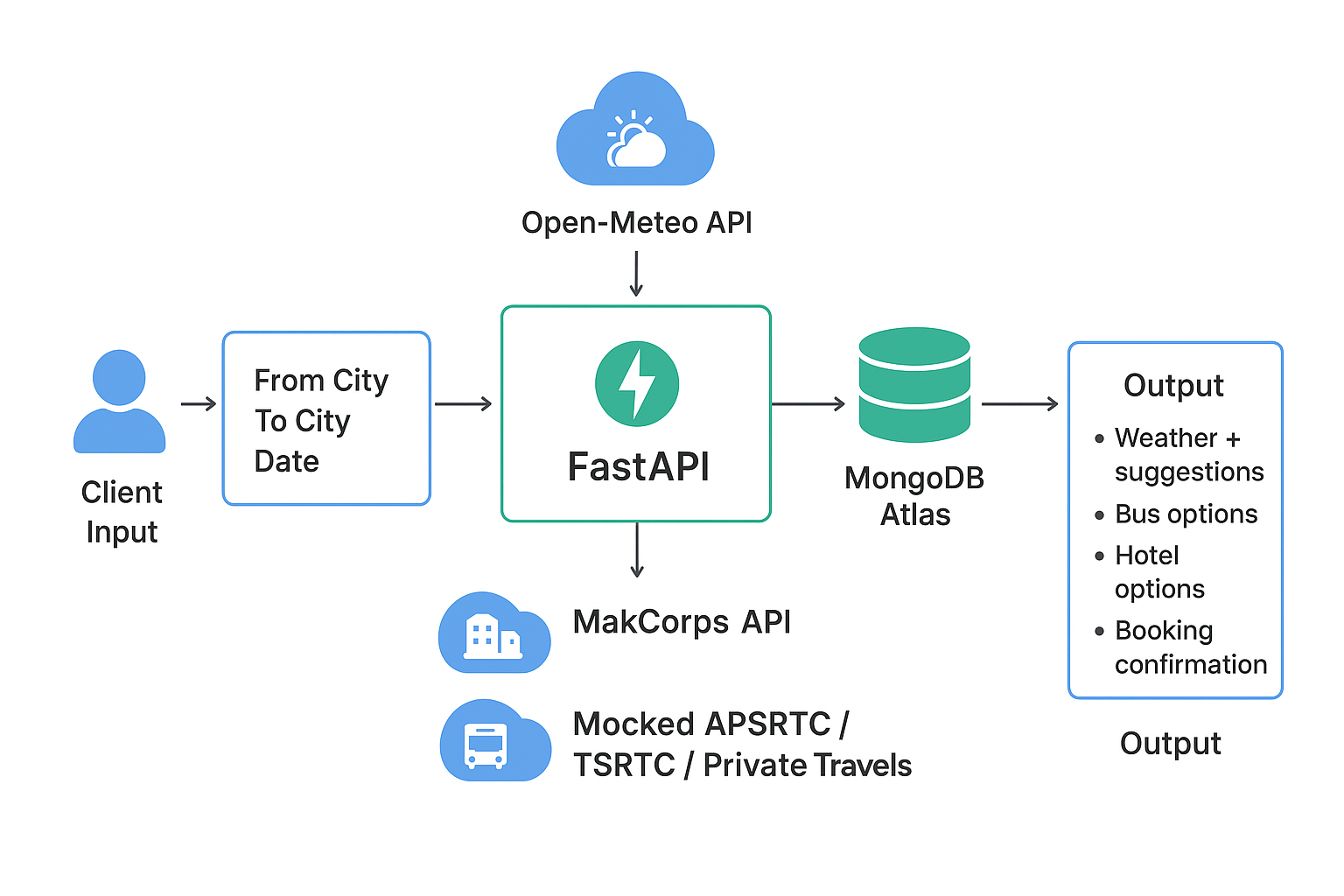
PTA addresses the challenge of fragmented travel planning by offering a unified backend:

* Plan trips with real-time weather and safety suggestions.
* Search and choose from APSRTC, TSRTC, and private buses.
* Explore hotels based on ratings and prices.
* Book both bus & hotel in one step.
* Store and retrieve bookings seamlessly from MongoDB Atlas.

# **3. System Architecture**

Components:

* FastAPI backend – API layer for search, booking, and retrieval.
* MongoDB Atlas – Cloud database for persistent bookings.
* Open-Meteo API – Provides weather forecasts.
* MakCorps API – Hotel listings.
* Mock Bus Service – Simulated APSRTC, TSRTC, and Private travels.



# **4. Workflow**

1. Search → Input from\_city, to\_city, depart\_date → Output: Weather + Buses + Hotels
2. Book → Choose bus + hotel, store in DB
3. Get Bookings → Retrieve all saved bookings

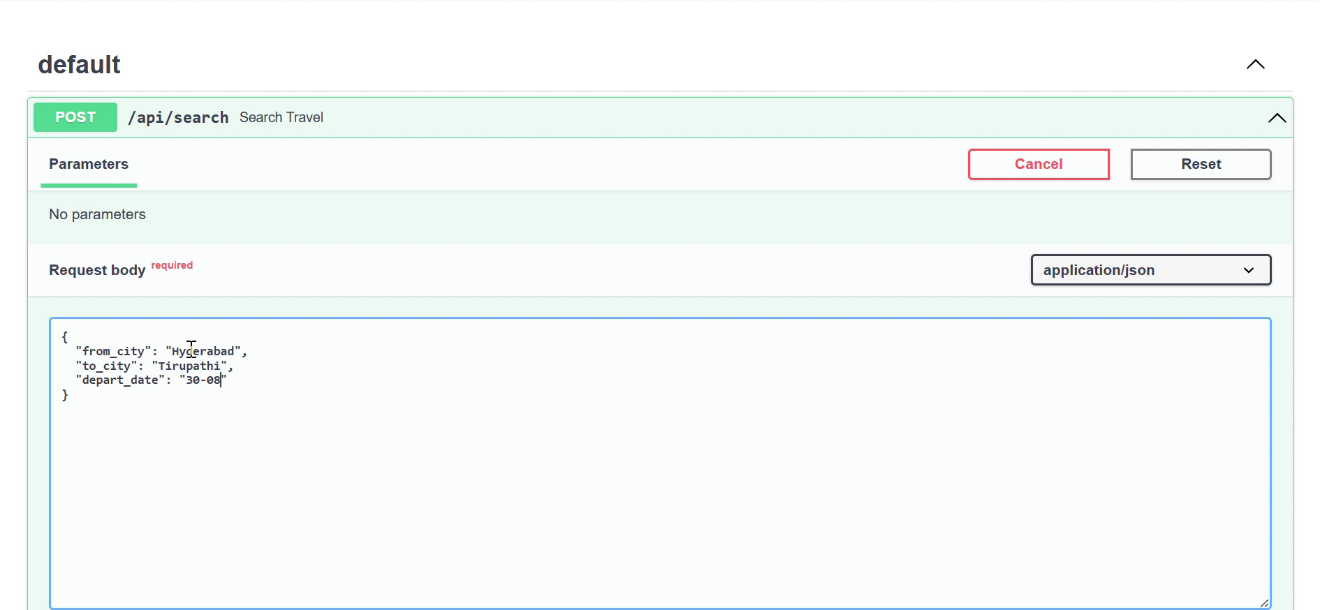
# **5. API Endpoints**

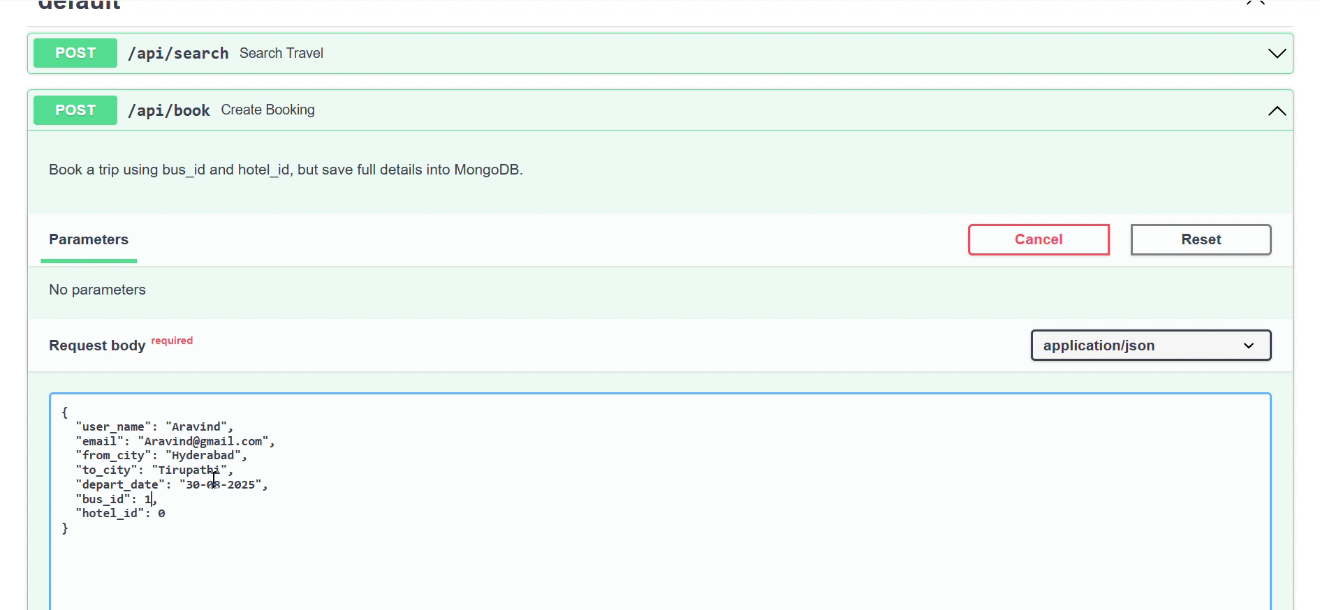
|  |  |  |
| --- | --- | --- |
| Method | Endpoint | Description |
| POST | /search | Search destination (weather, buses, hotels). |
| POST | /book | Book bus + hotel (saves to DB). |
| GET | /bookings | Retrieve all saved bookings. |

# **6. MongoDB Schema**

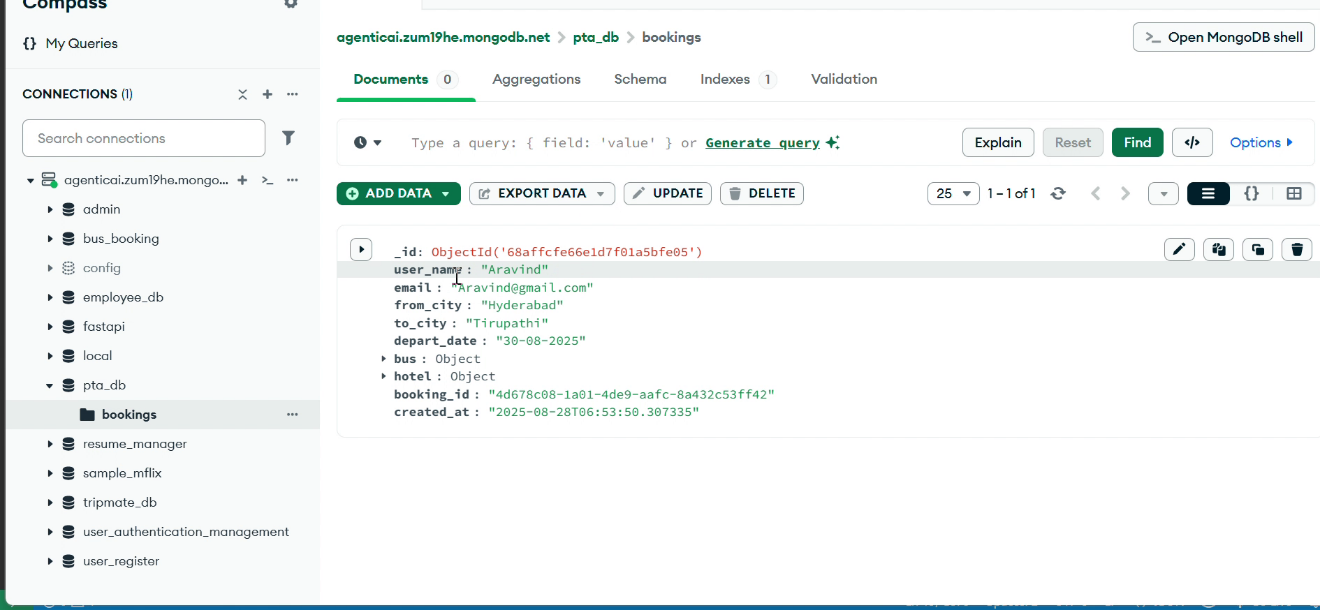
{  
 "\_id": ObjectId,  
 "user\_name": "string",  
 "email": "string",  
 "from\_city": "string",  
 "to\_city": "string",  
 "depart\_date": "string",  
 "bus": { "id": 1, "operator": "APSRTC" },  
 "hotel": { "id": 2, "name": "Grand Hotel" },  
 "created\_at": "timestamp"  
}

# **7. Endpoints Screenshots**

* **search travel**
* **Booking process**



* **Final bookings retrieval**



# **8. Sample JSON Response**

{  
 "count": 1,  
 "bookings": [  
 {  
 "user\_name": "Aravind",  
 "email": "p.aravind123@gmail.com",  
 "from\_city": "Hyderabad",  
 "to\_city": "Tirupathi",  
 "depart\_date": "2025-08-30",  
 "bus": {  
 "operator": "APSRTC Express",  
 "departure\_time": "2025-08-30 06:00",  
 "price": 500  
 },  
 "hotel": {  
 "name": "Vijayawada Grand Hotel",  
 "price\_per\_night": 2500,  
 "rating": 4.2  
 }  
 }  
 ]  
}

# **9. Deployment Notes**

- 🚀 Deploy on AWS / Azure (EC2, App Service, or containerized).  
- 🐳 Use Docker for portability and consistency.  
- ☁️ Database: MongoDB Atlas (cloud-managed, scalable).  
- 🔐 Secure APIs with authentication in production.

# **10. Conclusion**

✅ PTA is scalable, production-ready, and client-focused.  
✅ It integrates weather, buses, and hotels into one backend system.  
✅ Suitable for real-time travel planning and extendable for production use.  
  
📌 **Deliverables (included in repo):**  
- Full FastAPI + MongoDB source code (Personal\_Travel\_Assitant(PTA).  
- API documentation & Swagger UI.  
- Sample outputs (response\_1756364051907.json).  
- Demo video (Screen Recording 2025-08-28.mp4).  
- Professional PPT (for client presentation).