Where would the model be deployed?

To deploy our demand forecasting model into a production environment, we will leverage cloud services for scalability and reliability. Examples such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud . Alternatively, we can deploy the model on dedicated servers or containerization platforms like Docker if specific infrastructure requirements dictate.

API Design

The model will be accessible through a RESTful API designed using a web framework such as Flask, FastAPI, or Django. The API will feature endpoints for making predictions, with well-documented input data formats and expected response structures.

To accommodate the expected 10-20K client app requests every 15 minutes, we'll implement various strategies:

Load balancing will distribute incoming requests across multiple instances of our model deployed on different servers.

Auto-scaling features provided by cloud platforms will handle increased demand during peak times.

A content delivery network (CDN) will be employed for caching and delivering API responses more efficiently.

To make the model available for different geographies or areas within a city, we will set up a routing mechanism. This mechanism will intelligently direct incoming requests to the appropriate model instance based on the specified geography in the request. In cases of distinct geographic requirements, we will maintain separate model instances.

To ensure secure access to the API, we will implement token-based authentication. User roles and permissions will be defined to control who can access the API and what actions they can perform.

Logging and monitoring tools will track API usage, response times, and potential errors.

Alerting mechanisms will be set up to detect unusual behavior, increased error rates, or performance degradation.

System resource utilization will be continuously monitored to ensure the deployment environment can manage the expected load.