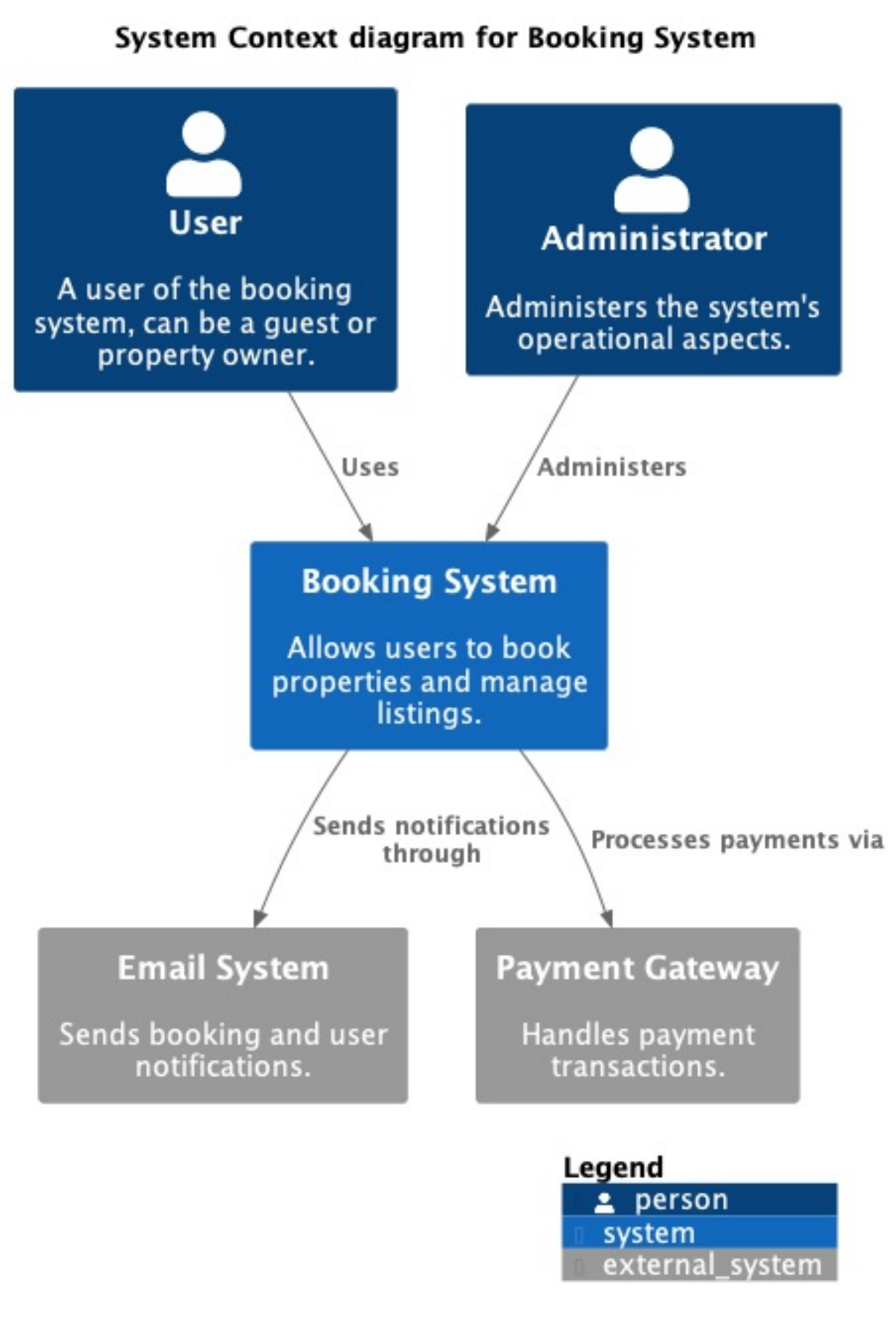
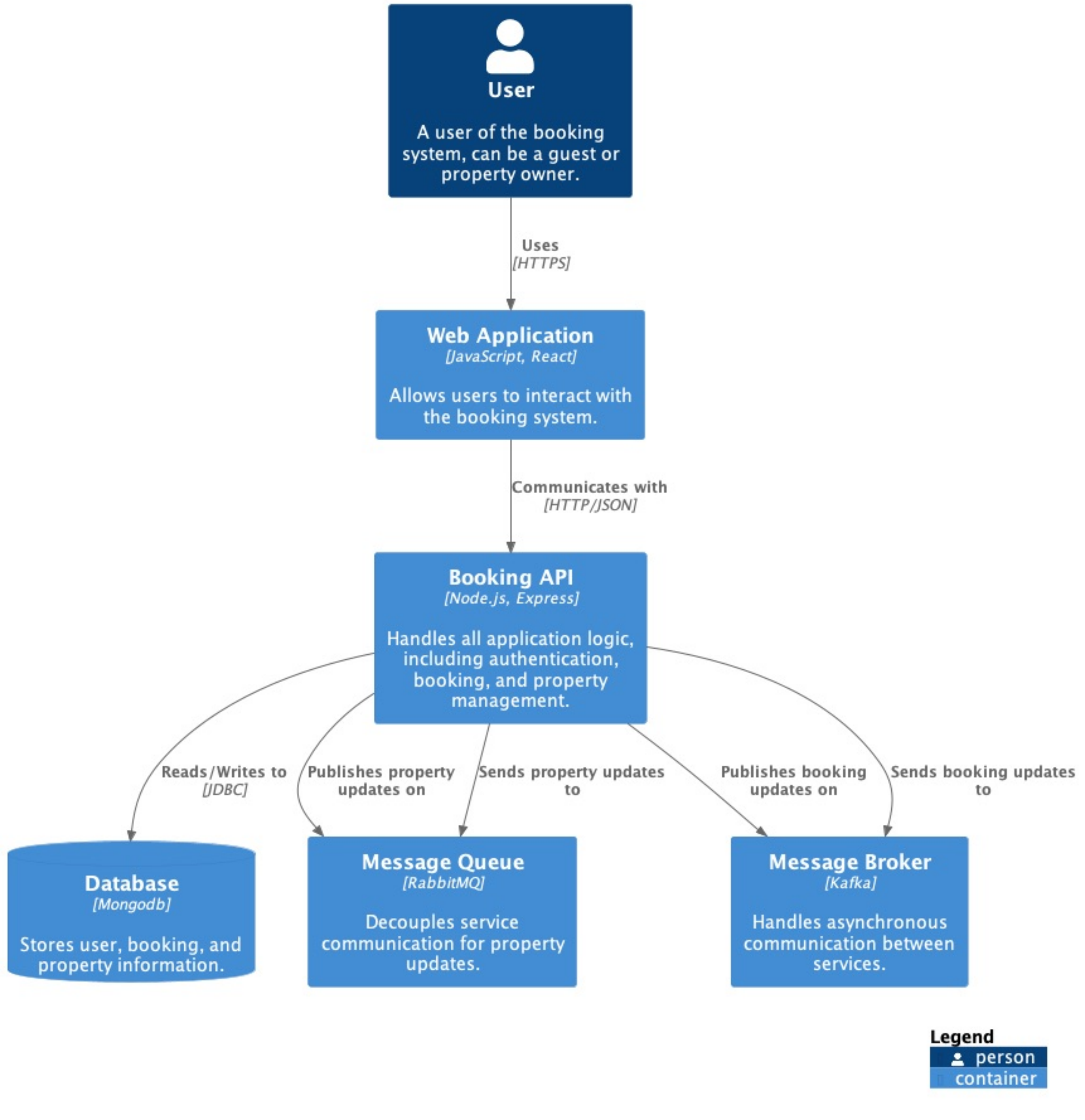
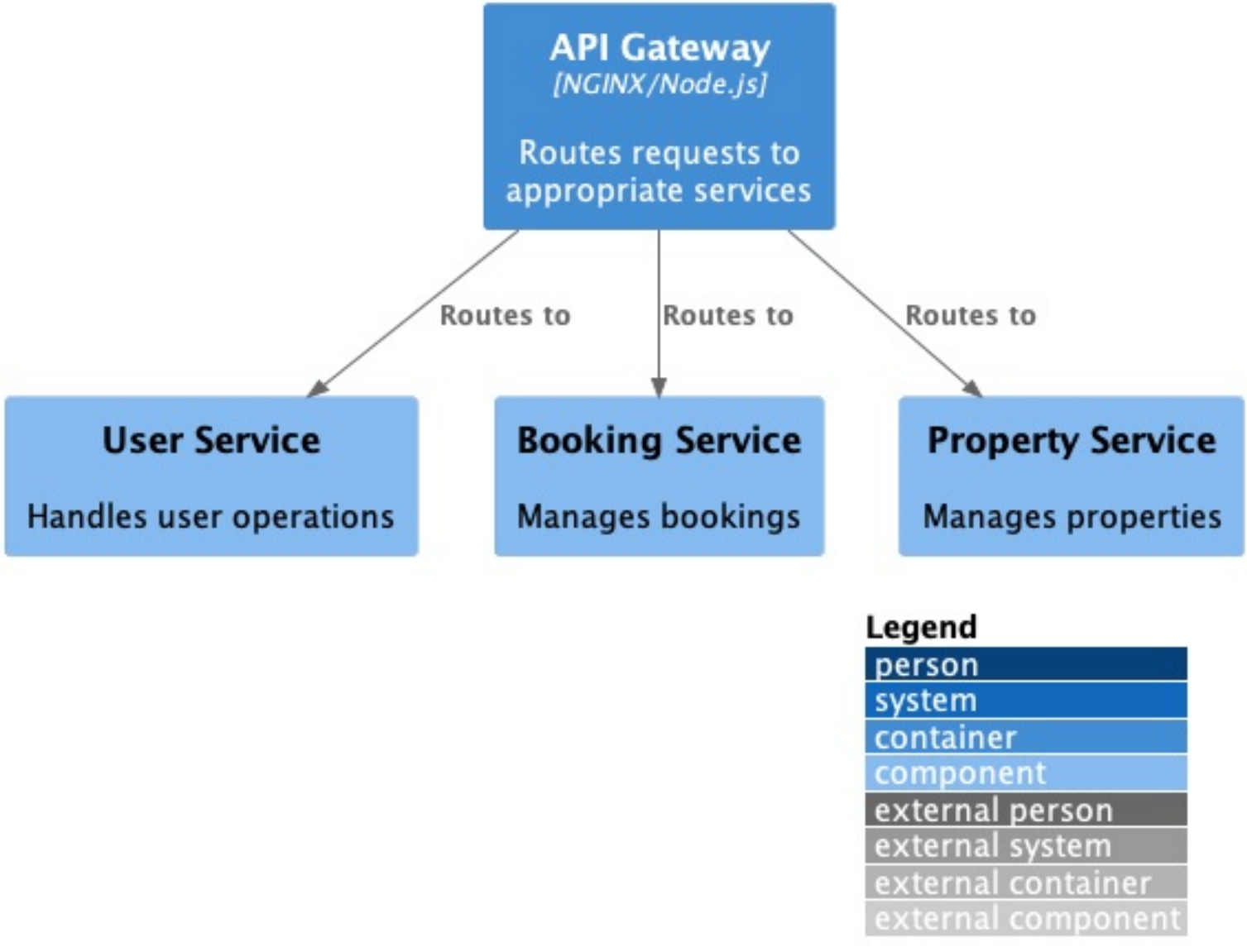
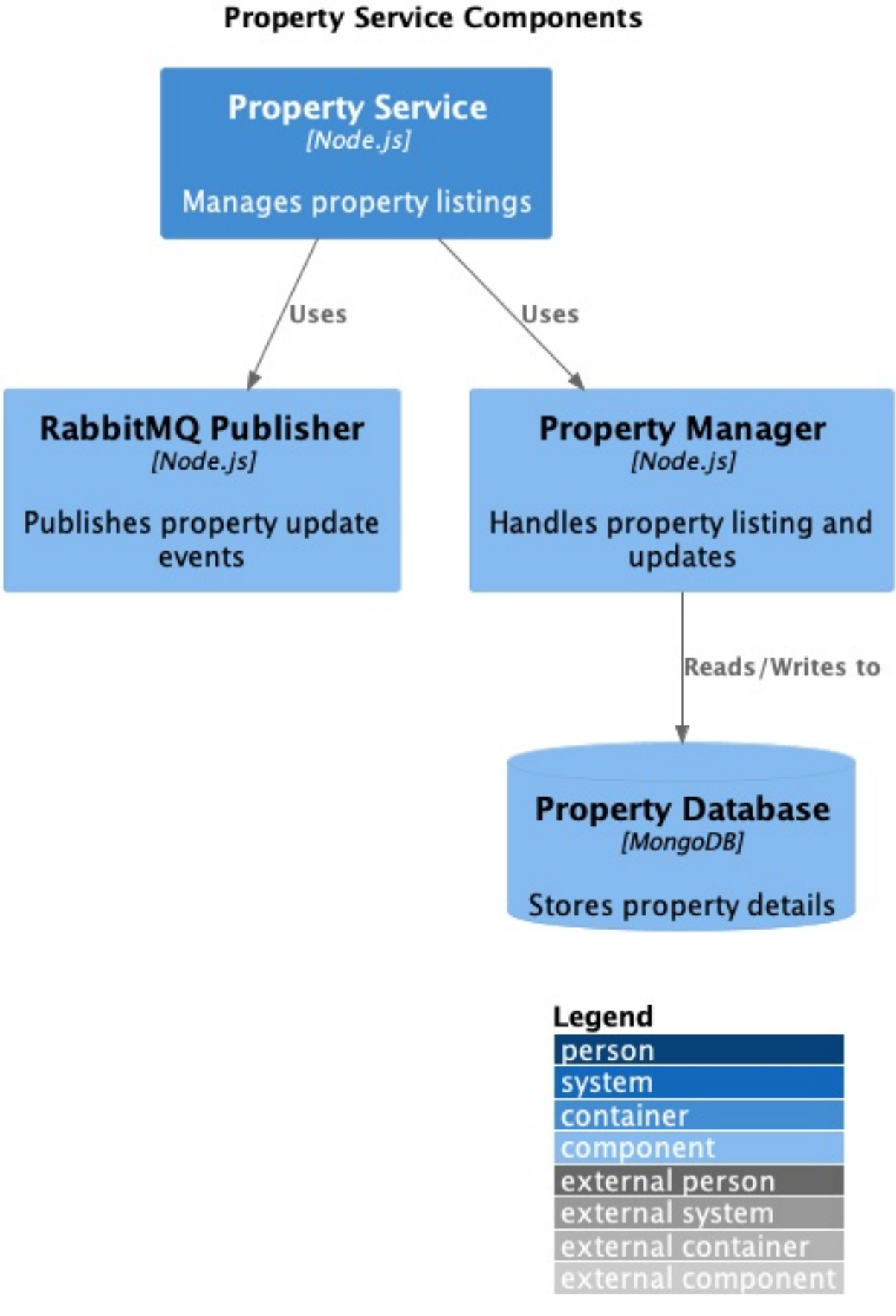

System Context diagram for Booking System
User
A user of the booking system, can be a guest or property owner.
Administrator
Administers the system's operational aspects.
Uses
Administers
Booking System
Allows users to book properties and manage listings.
Sends notifications through
Processes payments via
Email System
Sends booking and user notifications.
Payment Gateway
Handles payment transactions.
Legend
person
system
external_system

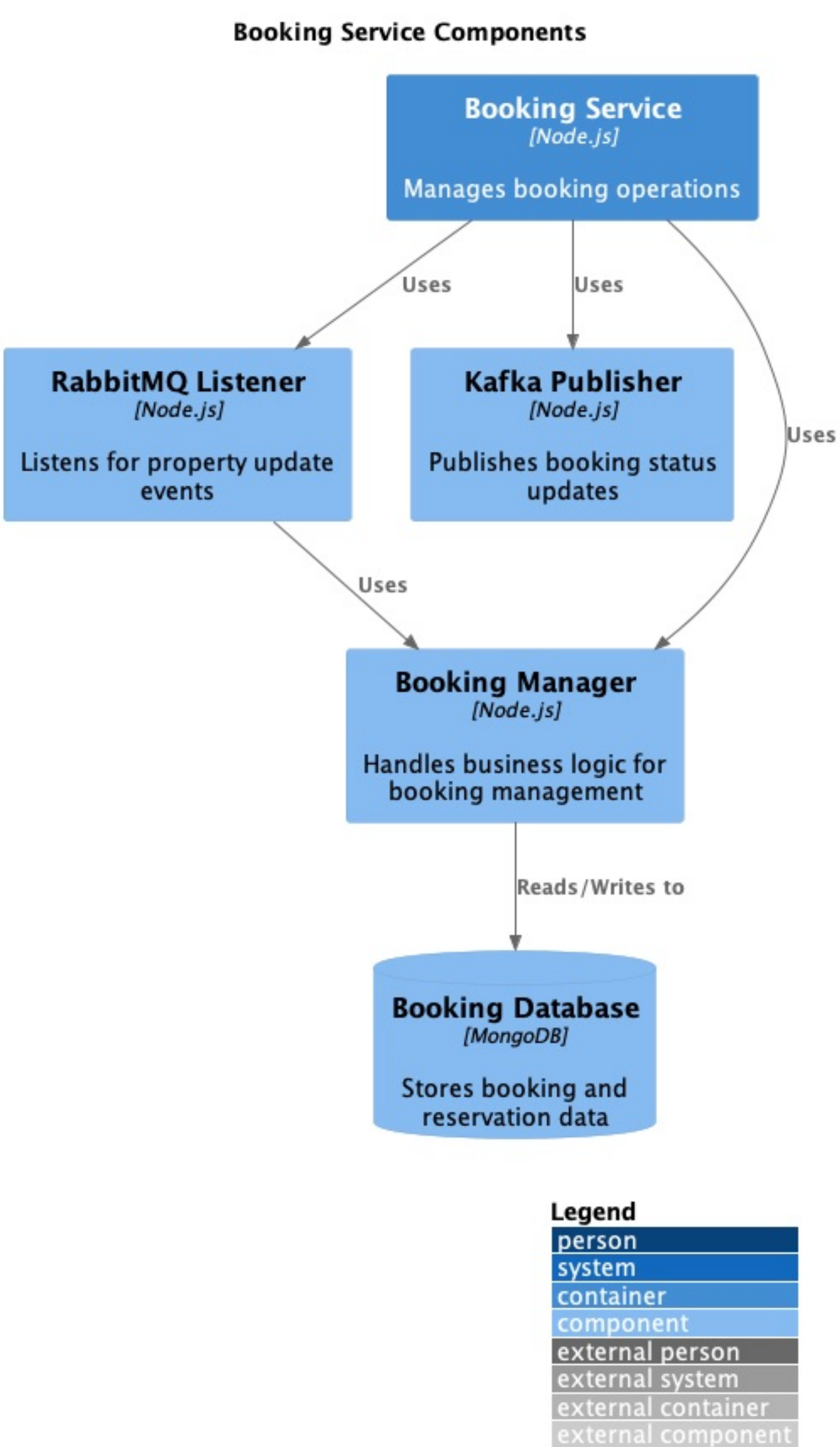
Container diagram for Booking System



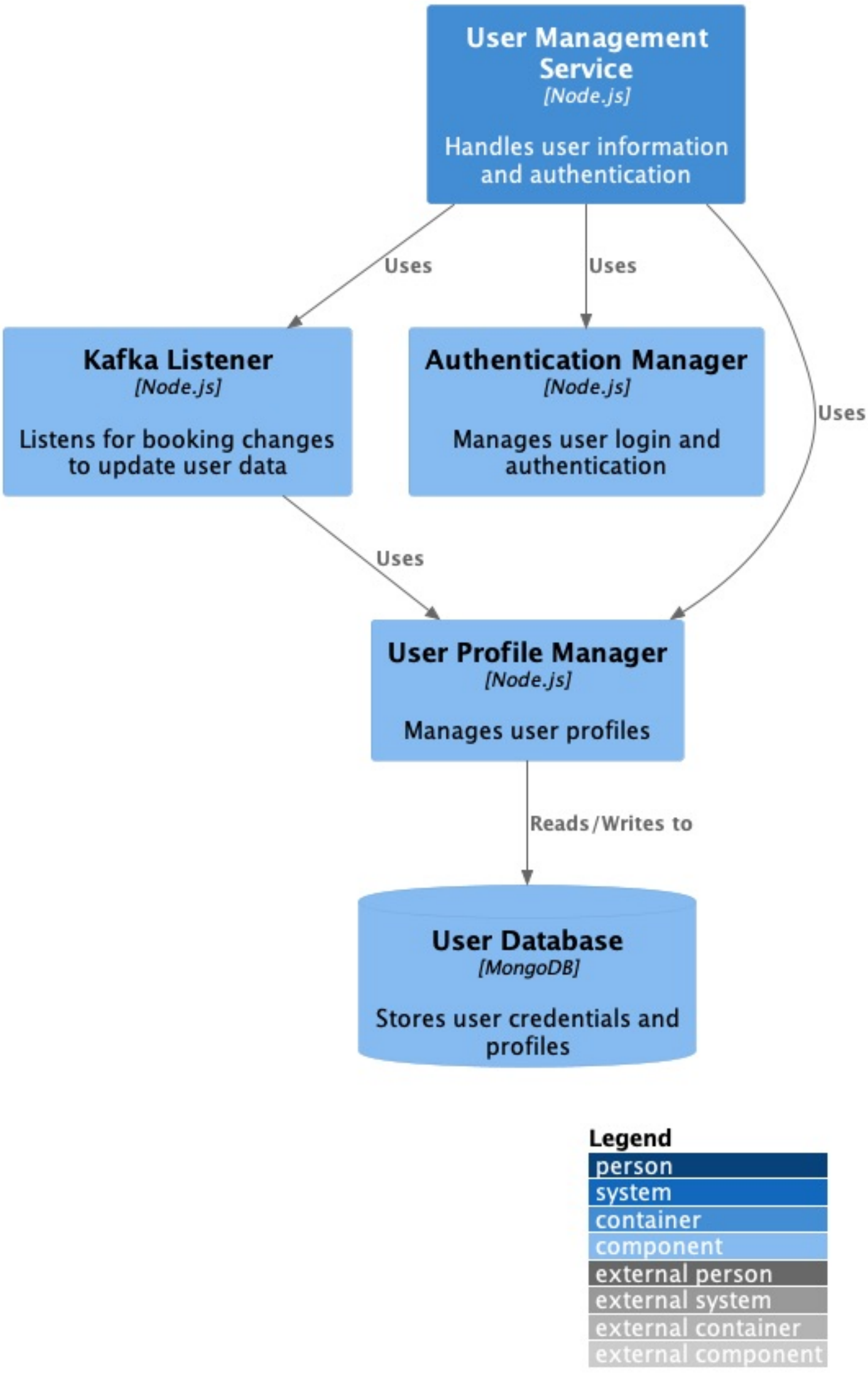
API Gateway Components




Property Service Components
Property Service
[Node.js]
Manages property listings
Uses
Uses
RabbitMQ Publisher
[Node.js]
Publishes property update events
Property Manager
[Node.js]
Handles property listing and updates
Reads/Writes to
Property Database
[MongoDB]
Stores property details
Legend
person
system
container
component
external person
external system
external container
external component


Booking Service Components
Booking Service
[Node.js]
Manages booking operations
Uses
Uses
Uses
RabbitMQ Listener
[Node.js]
Listens for property update events
Kafka Publisher
[Node.js]
Publishes booking status updates
Uses
Booking Manager
[Node.js]
Handles business logic for booking management
Reads/Writes to
Booking Database
[MongoDB]
Stores booking and reservation data
Legend
person
system
container
component
external person
external system
external container
external component

User Management Service Components



«container»
Frontend Application
[React]

Provides user interface for booking and managing properties.

Routes requests through

«container»
Server
[Nginx]

Routes requests to the appropriate service.

Routes booking requests

Routes property requests

Routes user requests

Booking System
[System]

«container»
Booking Service
[Node.js]

Manages and updates bookings.

«container»
Booking Database
[Stores booking details and statuses.]

Reads from and writes to

Publishes booking updates
[Kafka]

Sends property update notifications
[RabbitMQ]

Publishes property updates
[RabbitMQ]

«container»
RabbitMQ/Kafka
[Message broker for service communication.]

«container»
Property Database
[Stores property details and availability.]

Reads from and writes to

«container»
Property Service
[Node.js]

Manages property listings.

Consumes booking updates; Publishes user actions
[Kafka/RabbitMQ]

Sends user notifications
[Kafka]

«container»
User Management Service
[Node.js]

Handles user information and authentication.

«container»
User Database
[Stores user profiles and credentials.]

Reads from and writes to