Source Code: AI Generated Report

Generated on: 2026-02-07 16:30:06

```
from fastapi import FastAPI, HTTPException, Request
from fastapi.responses import JSONResponse
from pydantic import BaseModel, conlist
from sqlalchemy import create_engine
from sqlalchemy.orm import DeclarativeBase, Mapped, mapped_column
from sqlalchemy.orm import sessionmaker
import pandas as pd
from fastapi.middleware.cors import CORSMiddleware
from fastapi.middleware.gzip import GZipMiddleware
from fastapi_limiter import FastAPILimiter


# Database setup
DATABASE_URL = "sqlite:///:memory:"
engine = create_engine(DATABASE_URL)
SessionLocal = sessionmaker(autocommit=False, autoflush=False, bind=engine)


class Base(DeclarativeBase):
    pass


# Models
class Product(BaseModel):
    name: str
    price: float
```

```
class BillRequest(BaseModel):
    products: conlist(Product)

# FastAPI app
app = FastAPI()
app.add_middleware(CORSMiddleware,      allow_origins=["*"],      allow_credentials=True,
allow_methods=["*"], allow_headers=["*"])
app.add_middleware(GZipMiddleware)

# Rate limiting
FastAPILimiter.init(app, key="rate_limit", default_limits=["5/minute"])

@app.post("/calculate-bill")
@FastAPILimiter.limit("5/minute")
async def calculate_bill(bill_request: BillRequest):
    try:
        # Convert products to DataFrame
         df = pd.DataFrame([{"name": product.name, "price": product.price} for product in
bill_request.products])

        # Calculate total price
        total_price = df['price'].sum()

        # Calculate GST and final bill
        gst = total_price * 0.12
```

```
        final_bill = total_price + gst

        # Prepare response
        response = {
            "total_price": total_price,
            "gst": gst,
            "final_bill": final_bill,
            "breakdown": df.to_dict(orient="records")
        }
        return JSONResponse(content=response)

    except Exception as e:
        raise HTTPException(status_code=500, detail=str(e))

# Create database tables
Base.metadata.create_all(bind=engine)

# Testing
if __name__ == "__main__":
    from fastapi.testclient import TestClient

    client = TestClient(app)

    # Test with a valid list of products
    response = client.post("/calculate-bill", json={
        "products": [
```

```
        {"name": "Product A", "price": 100.0},
        {"name": "Product B", "price": 200.0}
    ]
})
print(response.json())

# Test with an empty product list
response = client.post("/calculate-bill", json={"products": []})
print(response.json())

# Test with invalid price type
response = client.post("/calculate-bill", json={
    "products": [
        {"name": "Product A", "price": "invalid_price"}
    ]
})
print(response.json())
```