Project: Quantum Compute Usage Optimization

Objective: Develop a system to minimize the overall cost of running workloads on Quantum Compute LLC's leased quantum computing resources (QPU blocks).

Key Components:

1. Web Interface: Design a web page to display summary data and provide a chat interface. The use of Next.js framework is preferred.
2. APIs: Develop APIs to interact with the system and retrieve data. The FastAPI framework is preferred.
3. LLM (Large Language Model): Utilize an LLM to answer interactive questions and provide insights.
4. Agentic AI: Implement an agentic AI framework (e.g., LangChain) to enable conversational interactions.
5. Data Analysis: Analyze simulated data for the past 6 months to optimize QPU block distribution and predict future allocations.

Tasks:

1. Simulate data for the past 6 months.
2. Optimize QPU block distribution to minimize total cost.
3. Predict future allocations using past data and usage patterns.
4. Develop a web page with summary data and a chat interface.
5. Implement LLM, agentic AI, and conversation chat components for interactive questions.

This project requires a multidisciplinary approach, combining data analysis, API development, web design, and AI implementation.

Situation:

A quantum computing startup is developing a model to lease its large quantum computing resources (Quantum Processing Units – QPU) to customers by dividing the computing into small blocks.

1. A QPU block can contain 1 – 100,000,000 QPUs.
2. There are three categories of QPU blocks, differentiated by cost structure:
   2.1. Atom
   2.2. Photon
   2.3. Spin
3. Once leased, the customer retains the QPU block, even if it is not used frequently.
4. The company plans to create new blocks regularly and expects the customers to use the newer blocks more frequently while the older blocks still remain in use.
5. Each compute block type has the following cost components:

Cost Categories	Description	Atom	Photon	Spin

Lease Fee	Once a QPU block is acquired, it is charged a lease fee for the rest of its life, even if it is idle.	$3.00 per hour per QPU Block	$1.50 per hour per QPU Block	$0.40 per hour per QPU Block
Acquisition Cost	One time cost to acquire a QPU Block	$0.20 per QPU Block	$0.20 per QPU Block	$0.20 per QPU Block
Workload Trigger Cost	Cost charged every time a workload is triggered (started) in a block	$0.01 per Workload trigger	$0.01 per Workload trigger	$0.01 per Workload trigger
Workload execution cost	Workload runtime cost (every time a workload is executed)	$0.01 per QPU Block	$0.05 per QPU Block	$0.20 per QPU Block
QPU Block Transfer Cost	Charged when a Block is moved to this type	$0.01 per QPU Block	$0.10 per QPU Block	$.25 per QPU Block

Rules for the customers to lease and use the QPU Blocks

1. Customer A needs to lease 1,000 -10,000 new QPU blocks daily (random number). A block can only be leased once, i.e., once leased, it gets added to the leased pool and can not be part of the future random number generation. A block can only be leased once and stays assigned to the customer once acquired (each of 100,000,000 blocks has a unique ID to track). The leased block can be of any type (Atom, Photon, or Spin).
2. The customer needs to run 1,000,000 – 50,000,000 workloads daily (pick a random number in this range). Each workload needs a QPU block to execute. Each workload will require a QPU block of the size of 1 – 100,000,000 QPU Units (assigned randomly). The workloads in the daily pool will be created such that it can fit in one of the leased QPU blocks so far (including the blocks leased today). See, number 3 below for additional information.
3. Once a pool of daily required workloads has been created, this should be randomly allocated to execute on the QPU blocks the customer has leased with the following considerations:
    a. 50-60% of the workloads should be assigned to the QPU blocks leases today.
    b. The remaining 40-50% should be assigned to the QPU blocks from the older pool (QPU blocks leased before today)
4. A QPU block can execute multiple workloads at a time, i.e., it is multithreaded. For example, multiple workloads with the exact QPU needs can run on a single QPU

block. Example – 10 workloads with 3 QPU units can run on a single quantum block in parallel.

Problem:

The objective is to develop a system for the customer to minimize the overall cost of running the workloads.

Your Tasks:

1. Create simulated data for the past 6 months (using above information and rules), assuming the pool of QPU blocks leased at the end of 6 months is equally distributed between Atom, Photon, and Spin categories.
    a. The data should include the date, number of QPU blocks leased (distributed equally between 3 categories, workloads executed on each leased QPU block, and total daily cost.
2. Optimize distribution of leased blocks to minimize the total cost. Transfer blocks between categories as needed.
3. Using past data and usage patterns, predict allocating leased blocks by category by optimizing for total customer cost.
4. Create a web page for:
    a. Show summary data (i.e. QPU blocks leased, average workloads executed per block, number of workloads every day) etc.
    b. A chat interface using an LLM answering the questions like:
        i. Tell me the top 10 most active QPU blocks (by number of workloads executed),
        ii. What will happen to the cost if I only use Atom blocks?
        iii. Generate a graph to show the trend of daily costs.
5. Use LLM, an agentic AI framework like LangChain and conversation chat components for interactive questions.

Notes:

1. Each workload should require a specific QPU config (number of units) and can only be run on the exact matching size QPU block.
2. Multiple workloads of the same size can run on the same QPU block.