# Task 1.2 – Cloud-Based RAG Assistant Cost Estimation

To build a Retrieval-Augmented Generation (RAG) assistant on the cloud, the main cost factors are:

- Total number of input-output tokens generated daily.
- Model pricing policy: more powerful models have a higher cost per million tokens.
- Vector databases (DB) for storing and retrieving embeddings.

For small to mid-sized companies, the main driver of costs is typically the model choice.

## Assumptions

$$\begin{aligned}
\text{Total queries per month} &= Q \in [3{,}000, 30{,}000] \\
\text{Average input tokens per query} &= T_{\text{in}} = 1{,}400 \\
\text{Average output tokens per query} &= T_{\text{out}} = 600 \\
\text{Input/Output token split} &\approx 70/30 \\
\text{Vector DB, Weaviate Serverless:} &= \$25/\text{month} + \$0.095 \text{ per 1M vector dimensions}
\end{aligned}$$

## Pricing for Major LLMs (per 1M tokens)

| Model | Input ($/1M tokens) | Output ($/1M tokens) |
|---|---|---|
| GPT-5 | 1.25 | 10.0 |
| GPT-5 Mini | 0.25 | 2.0 |
| GPT-5 Nano | 0.05 | 0.40 |
| GPT-4o | 2.50 | 10.0 |
| GPT-4o Mini | 0.15 | 0.6 |
| o1 Pro | 150 | 600 |
| Claude Sonnet 4 | 3.00 | 15.0 |
| Claude Opus 4 | 15.0 | 75.0 |
| Claude Haiku 3.5 | 0.80 | 4.0 |
| Gemini 1.5 Pro | 1.25 | 5.0 |
| Cohere Command R+ | 2.50 | 10.0 |

## Cost Formula

The total monthly cost $C_{\text{total}}$ of operating a RAG assistant is:

$$\boxed{C_{\text{total}} = Q\left(T_{\text{in}} \cdot P_{\text{in}} + T_{\text{out}} \cdot P_{\text{out}}\right) + C_{\text{emb}} + C_{\text{DB}}} \tag{1}$$

Where:

$$
\begin{aligned}
Q &= \text{Total number of queries per month} \\
T_{\text{in}} &= \text{Average number of input tokens per query} \\
T_{\text{out}} &= \text{Average number of output tokens per query} \\
P_{\text{in}} &= \text{Price per input token for the chosen LLM (per token)} \\
P_{\text{out}} &= \text{Price per output token for the chosen LLM (per token)} \\
C_{\text{emb}} &= \text{Embedding cost (negligible)} \\
C_{\text{DB}} &= \text{Managed vector database monthly cost}
\end{aligned}
$$

### Example Calculations for GPT-4o Mini + Weaviate Serverless

**Small company:**

$$
\begin{aligned}
C_{\text{input}} &= Q_{\text{small}} \cdot T_{\text{in}} \cdot P_{\text{in}} = 3{,}000 \cdot 1{,}400 \cdot 0.00000015 \approx 0.63 \text{ USD} \\
C_{\text{output}} &= Q_{\text{small}} \cdot T_{\text{out}} \cdot P_{\text{out}} = 3{,}000 \cdot 600 \cdot 0.0000006 \approx 1.08 \text{ USD} \\
C_{\text{LLM}} &= 0.63 + 1.08 = 1.71 \text{ USD} \\
C_{\text{total}} &\approx 1.71 + 26 \approx 27.71 \text{ USD/month}
\end{aligned}
$$

**Large company:**

$$
\begin{aligned}
C_{\text{input}} &= 30{,}000 \cdot 1{,}400 \cdot 0.00000015 \approx 6.3 \text{ USD} \\
C_{\text{output}} &= 30{,}000 \cdot 600 \cdot 0.0000006 \approx 10.8 \text{ USD} \\
C_{\text{LLM}} &= 6.3 + 10.8 = 17.1 \text{ USD} \\
C_{\text{total}} &\approx 17.1 + 27 \approx 44.1 \text{ USD/month}
\end{aligned}
$$

# Local-Based RAG Assistant Deployment

To deploy a Retrieval-Augmented Generation (RAG) assistant locally, both model selection and hardware requirements must be considered. The efficiency and cost of the system will scale with the size of the chosen model and the infrastructure needed to support it.

## Model Options

**Small company (lightweight models, 7B–32B parameters):**

- LLaMa-2/3 (7B–13B)
- Mistral 7B
- Zephyr 7B
- Vicuna 7–13B
- Phi-3 Medium (14B)
- Deepseek-R1-Distill-Qwen-32B

**Mid to large company (32B–120B parameters):**

- LLaMa-2/3 70B
- Mistral 8x7B (Mixture-of-Experts)
- Falcon 40B
- OpenAI GPT-oss-20B
- OpenAI GPT-oss-120B

**State-of-the-art large reasoning models (180B+):**

- DeepSeek-R1 (671B)
- Falcon 180B

## System Components

To deploy a RAG system, two main subsystems are required:

1. **Vector Database (e.g., Weaviate)** – used for embeddings and retrieval. Costs here are negligible.
2. **Inference Hardware** – the primary cost driver:
   - **GPU:** Executes model inference. Larger models require multiple GPUs, which form the main bottleneck.
   - **CPU:** Orchestrates preprocessing, retrieval, and concurrent requests.
   - **RAM:** Provides temporary working memory for the CPU and helps with parallel queries.
   - **SSD/NVMe:** Stores model weights, document embeddings, and user query history.

## Cost Estimates

**Small company setup (7B–32B models):**

- GPU workstation/server:
  - 1× RTX 4090 system: \$5,000–\$7,000
  - OR 1× A100 system: \$12,000–\$15,000
- Other infrastructure (CPU, RAM, storage): \$2,000–\$3,000

$$\text{Total Initial Cost} \approx \$7,000\text{–}\$15,000$$

**Large company setup (32B–120B models):**

- Multi-GPU server:
  - 4× A100 80GB system: \$90,000–\$110,000

  - OR 4× H100 system: \$140,000–\$160,000

- Other infrastructure (CPU, RAM, storage): \$10,000–\$15,000

$$\text{Total Initial Cost} \approx \$100,000\text{–}\$160,000$$

**Operational costs:** An additional 10–20% of hardware costs should be allocated annually for electricity, cooling, and maintenance.

## Advantages of Local Deployment

- Strong data privacy: company documents remain on internal servers.
- Full control over model selection and deployment.
- Independence from external providers (offline availability).
- Predictable and upfront hardware costs.

## Disadvantages of Local Deployment

- High upfront costs that scale with model complexity.
- Ongoing maintenance burden (hardware monitoring, system updates).
- Electricity and cooling overhead.
- Hardware obsolescence and model degradation risks over time.

## Conclusion

For small companies with limited databases, local deployment is a realistic option using smaller models on modest hardware. For large enterprises, however, local deployment becomes challenging to scale. It is feasible only when data privacy and regulatory compliance outweigh the convenience, elasticity, and lower operational burden of cloud-based solutions.