Gap Analysis: GenAI Engineer

John Huber's profile demonstrates extensive experience and a strong skill set as a UX Designer, with 7+ years delivering web and mobile products. His expertise lies in user research, wireframing, prototyping, UI design, A/B testing, and utilizing design tools like Adobe Suite, Sketch, and InVision. However, the GenAI Engineer role is highly technical and demands deep proficiency in advanced programming (Python, C++), deep learning frameworks (PyTorch, TensorFlow), generative AI architectures (LLMs, Transformers, Diffusion Models), MLOps, cloud AI services, data engineering for AI, model optimization, and responsible AI. John's background lacks any of the core technical qualifications and hands-on experience required for this specialized engineering position, indicating a fundamental and comprehensive skill gap.

Missing Skills:

- Python (including NumPy, Pandas, Scikit-learn)
- C++
- PyTorch
- TensorFlow/Keras
- Transformer architectures
- Large Language Models (LLMs) (GPT, Llama, Mixtral, Falcon, fine-tuning, pre-training, adapting)
- Diffusion Models (Stable Diffusion, DALL-E)
- Generative Adversarial Networks (GANs) / Variational Autoencoders (VAEs)
- Advanced prompt engineering techniques
- Retrieval Augmented Generation (RAG) systems
- Agentic frameworks (LangChain, LlamaIndex)
- MLOps principles and tools (MLflow, Kubeflow, Sagemaker MLOps)
- Containerization (Docker)
- Orchestration (Kubernetes)
- Building and deploying RESTful APIs for model inference (FastAPI, Flask)
- Model serving frameworks (Triton Inference Server)
- Cloud provider's AI/ML services (AWS SageMaker, Google Cloud Vertex AI, Azure Machine Learning)
- Cloud compute resources (EC2, GCE, AKS)
- Data preprocessing, cleaning, and feature engineering for large, unstructured datasets
- Vector databases (Pinecone, Weaviate, Chroma, Qdrant)
- Data pipelines and ETL tools
- Model quantization, pruning, and distillation techniques
- GPU optimization (CUDA, cuDNN) and distributed training frameworks
- Evaluating generative model outputs (perplexity, BLEU, ROUGE)
- Responsible AI principles (bias detection, fairness, privacy, safety)
- Red teaming and alignment