# Medical Coding AI Prototype: Hardware Specifications

This document details the recommended and essential hardware specifications for developing the Medical Coding AI hackathon prototype. These specifications are crucial for ensuring efficient development, especially given the computational demands of transformer-based NLP models.

---

## 1. Essential Hardware Specifications

These are the core components that will significantly impact your development experience and the performance of your prototype.

- **CPU (Central Processing Unit):**
    - **Recommendation:** Intel Core i5/i7 (10th Gen or newer) or AMD Ryzen 5/7 (3000 series or newer).
    - **Reasoning:** A modern, multi-core CPU is vital for overall system responsiveness, running Python scripts, handling data pre-processing, and managing the development environment efficiently.
- **RAM (Random Access Memory):**
    - **Minimum:** 16 GB
    - **Recommended:** 32 GB
    - **Reasoning:** Transformer models are memory-intensive. Loading large pre-trained models, processing batches of data, and running multiple development tools concurrently can quickly consume available RAM. 32GB provides a much smoother experience, reducing the likelihood of memory-related bottlenecks.
- **GPU (Graphics Processing Unit):**
    - **Minimum:** NVIDIA GeForce RTX 2060 (6GB VRAM) or equivalent.
    - **Recommended:** NVIDIA GeForce RTX 3060 (12GB VRAM) or better (e.g., RTX 3070, 3080, 4070, 4080).
    - **VRAM (Video RAM):** The amount of dedicated memory on the GPU is critical. More VRAM allows for larger models, bigger batch sizes during inference (and potential limited fine-tuning), and faster processing.
    - **Reasoning: This is the most critical component for deep learning tasks.** Running transformer models for inference and any form of fine-tuning is orders of magnitude faster on a GPU compared to a CPU. NVIDIA GPUs are highly preferred due to their robust CUDA platform support, which is widely adopted by popular deep learning frameworks like PyTorch and TensorFlow.
- **Storage:**
    - **Type:** SSD (Solid State Drive) is **essential**.
    - **Minimum Size:** 256 GB (usable space after OS and essential software).
    - **Recommended Size:** 512 GB or 1 TB SSD.

    - **Reasoning:** SSDs offer significantly faster read/write speeds compared to traditional Hard Disk Drives (HDDs). This translates to quicker loading times for large pre-trained models, faster data access, and a more responsive development environment overall. Transformer models and their associated data can occupy several gigabytes.

- **Operating System:**

    - **Recommendation:** Ubuntu 20.04+ (Linux) or Windows 10/11 (with Windows Subsystem for Linux 2 - WSL2 for a Linux development environment).
    - **Reasoning:** Deep learning development environments often have better native support, easier driver installation, and more straightforward library configurations on Linux. Windows with WSL2 provides an excellent compromise, allowing developers to leverage a full Linux environment for their ML work while still using Windows for other tasks.

---

## 2. Software/Driver Prerequisites

Beyond the hardware, specific software components are necessary to enable GPU acceleration for your deep learning tasks.

- **NVIDIA GPU Drivers:** Ensure you have the latest stable drivers installed for your NVIDIA GPU. These drivers facilitate communication between your operating system and the GPU.
- **CUDA Toolkit:** This is NVIDIA's parallel computing platform and API model. You need to install the CUDA Toolkit version that is compatible with your specific NVIDIA GPU drivers and the versions of PyTorch or TensorFlow you are using.
- **cuDNN:** The CUDA Deep Neural Network library (cuDNN) is a GPU-accelerated library for deep neural networks. It works in conjunction with CUDA to provide highly optimized routines for standard deep learning operations, significantly speeding up model training and inference.

---

**Summary:**

Prioritizing an NVIDIA GPU with sufficient VRAM and ample RAM will provide the most significant performance benefits for your Medical Coding AI prototype. While other components are important, these two are paramount for handling the computational demands of modern NLP models.