**2.3 User Classes and Characteristics**

Crear: Contiene los ids de los usuarios y de los cursos.

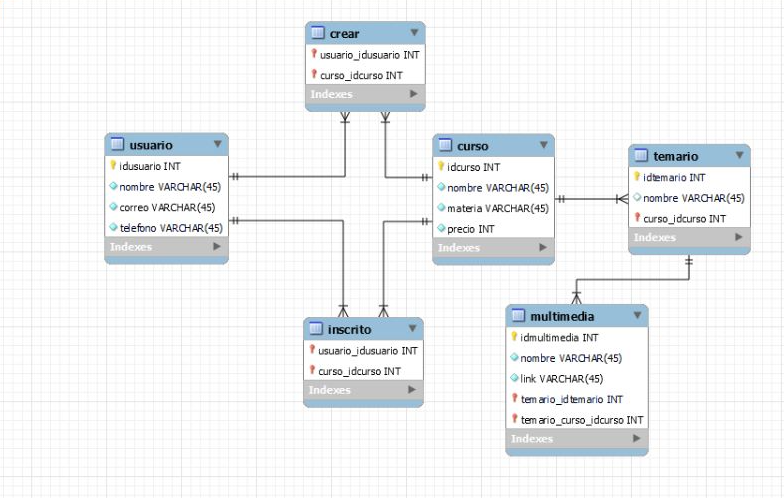
Usuario: Contiene la id del usuario junto con su nombre, teléfono y correo.

Inscrito: Contiene los ids de los usuarios y de los cursos.

Curso: Contiene la id del curso junto con su nombre, materia, precio.

Temario: Contiene la id del temario junto con su nombre y la id del curso.

Multimedia: Contiene la id del multimedia, su nombre, su link junto con la id del temario y la id del curso



**2.4 Operating Enviroment**

Todo se desarrolló en Django y se hosteo en una web.

**2.5 Design and implementation constraints**

1. Tiene que seguir siendo desarrollada en Django ya que usamos muchas características del lenguaje para diseñar el proyecto
2. La IP tiene que ser redirigida a un host externo y tiene que tener la capacidad mínima de 10 gbs para la base de datos.
3. Se recomendaría tener pocos cursos agregados para que los tiempos de espera sean adecuados, esto debido a la forma en que esta hecho la aplicación, optimiza el diseño y la forma grafica a costa de la velocidad de respuesta.

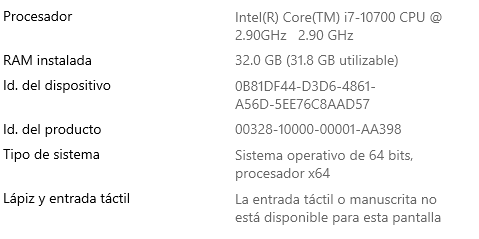
**2.6 Asumptions and Dependencies**

Esta aplicación no esta asociada con el ITAM, los cursos son impartidos por alumnos en salones previamente pedidos a la institución.

3.1 User Interfaces

**3.2 Hardware Interfaces**

Se utiliza solamente una computadora para un host temporal local con las siguientes especificaciones:



**3.4 Software Interfaces**

**HOST**: AWS Lambda

**Lenguaje**: Django versión 2.1

**IDE**: Visual Studio Code versión 1.73

**Ejecutable**: Google Chrome o Cualquier navegador

**3.5 Communications Interfaces**

Se utiliza una conexión desde Visual Studio Code hacia el navegador predeterminado del usuario para empezar a correr la aplicación. Además todas las solicitudes hacia la aplicación para ver cursos y agregar cursos los maneja los servicios de Amazon (AWS)