1. ¿Qué es UML y para qué se utiliza?

Es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo del software. Se utiliza para representar visualmente de forma eficiente la complejidad de un sistema u organización en un reducido número de diagramas

2. Nombrar y describir los distintos tipos de diagramas UML

* Diagramas de casos de uso: representan a los actores y casos de uso (procesos principales) que intervienen en un desarrollo de software.
* Diagramas de clases: para UML una clase es una entidad, no una clase software. Un diagrama de clases UML puede ser un diagrama del dominio o representación de conceptos que intervienen en un problema, o también un diagrama de clases software. El sentido de un diagrama UML se lo da la persona que lo construye.
* Diagramas de secuencia: suelen usarse para representar objetos software y el intercambio de mensajes entre ellos, representando la aparición de nuevos objetos de izquierda a derecha.
* Diagramas de colaboración: suelen usarse para representar objetos o clases y la forma en que se transmiten mensajes y colaboran entre ellos para cumplir un objetivo.
* Diagramas de estados: suelen usarse para representar cómo evoluciona un sistema (cómo va cambiando de estado) a medida que se producen determinados eventos.

3. ¿Cuáles son las ventajas que ofrece?

Sus ventajas son:

* Se puede usar para distintos tipos de sistemas
* Es fácil de entender
* Mejor soporte de a la planeación y al control de proyectos

4. ¿Cuáles son los elementos estructurales que contiene? Describir cada uno de ellos

Los elementos estructurales que contiene son:

* Clase: es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semánticas. Gráficamente se presenta como un rectángulo que normalmente contiene su nombre, atributos y operaciones
* Interfaz: es una colección de operaciones que especifican el servicio de una clase o componente. Puede representar el comportamiento completo de una clase o solo una parte de ese comportamiento. Una interfaz define las operaciones pero no la implementación de las operaciones. Se representa como un círculo
* Casos de uso: es una descripción de un conjunto de secuencias de acciones que un sistema que ejecuta y produce un resultado de interés para un actor particular. Se usa para estructurar aspectos de comportamiento de un modelo. Se representa como una eclipse incluyendo el nombre del caso de uso
* Componente: es una parte física y reemplazable de un sistema que conforma un conjunto de interfaces y proporciona la implementación de dicho conjunto. Un componente representa el empaquetamiento físico de diferentes elementos lógicos como clases, colaboraciones e interfaces. Se representa con un rectángulo con pestañas
* Nodo: es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional, por lo general dispone de algo de memoria y capacidad de procesamiento. Se utiliza para modelar la topología del hardware en el que se ejecuta el sistema. Se representa como un cubo.

5. Explique con sus palabras qué entiende por orientación a objetos

La orientación a objetos es una forma de programar objetos de la vida real en el entorno virtual, o sea pasar un objeto como por ejemplo: perro, a código. Entonces tendríamos una clase (objeto ) llamada perro, con atributos (características del objeto): marrón y bajo, con métodos (las acciones del objeto): comer, tomar y correr.