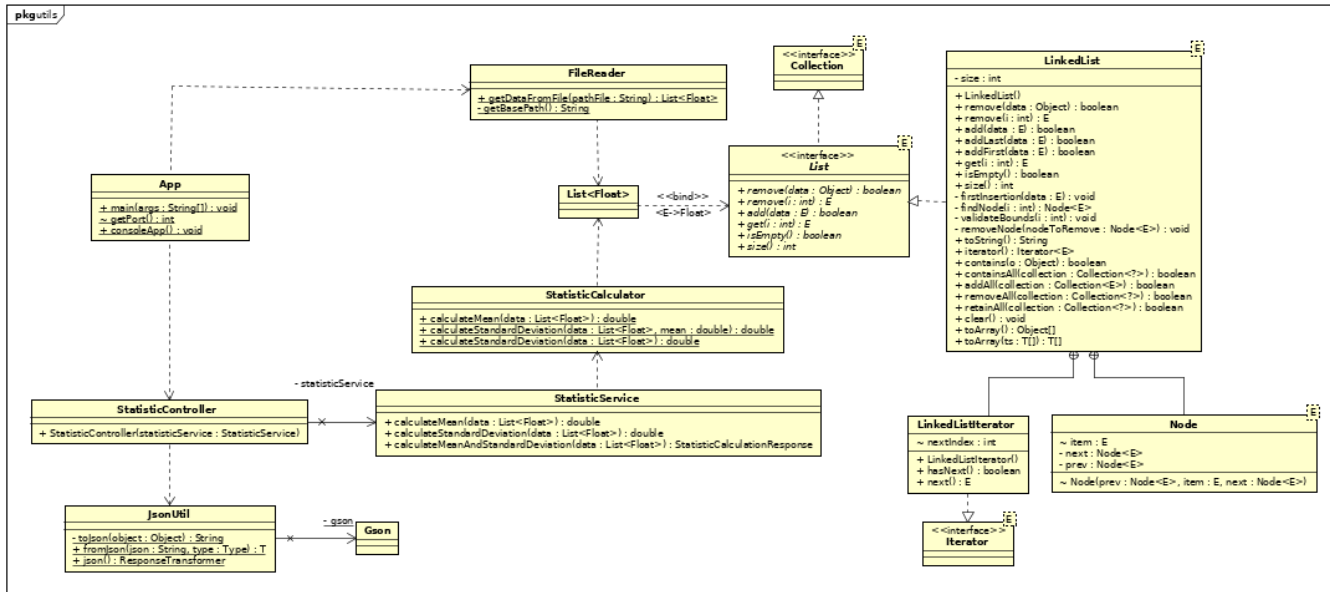


## Descripcion del diseño

### Diagrama de clases UML



- **LinkedList**: permite almacenar los datos de entrada, provee mecanismos de inserción, remoción y acceso rápido para los elementos en los extremos de la lista (cabeza y cola), para su funcionamiento interno se usa la Clase Node.
- **Node**: Describe un nodo de la estructura de datos LinkedList, almacena el valor en el nodo y la referencia del siguiente y anterior nodo en la lista.
- **LinkedListIterator**: Describe un iterador de la clase LinkedList, itera desde first hasta last.
- **List**: Interfaz que describe el comportamiento de una lista estándar.
- **StatisticCalculator**: Provee métodos para realizar cálculos estadísticos (media y desviación estándar).
- **StatisticController**: Se encarga de manejar la lógica asociada a http, REST y delega la petición en un contexto Java a el servicio.
- **StatisticService**: Se encarga de realizar e interpretar peticiones estadísticas.
- **FileReader**: Se encarga de leer archivos y entregarlos como objetos Java.
- **App**: Se encarga de orquestar a las demás clases para la lectura del archivo, cálculos estadísticos y dar la salida al usuario o levantar el API.