

Contratos de las operaciones:

Consultar datos:

Sección descargar datos:

Recolectar datos:

Contrato código: Nodo

Operación: Nodo()

Referencias: Recolectar datos

Precondiciones:

Postcondiciones: Devuelve una instancia de nodo.

Contrato código: establecerConfiguracion

Operación: establecerConfiguracion()

Referencias: Recolectar datos

Precondiciones: Es necesario tener un archivo de configuraciones.

Postcondiciones: Guarda las configuraciones, en formato de diccionario.

Contrato código: iniciarTiempo

Operación: iniciarTiempo()

Referencias: Recolectar datos

Precondiciones: Debió haber guardado las configuraciones.

Postcondiciones: Guarda en formato "data" la hora y minuto en que inició el programa.

Contrato código: Instrumento

Operación: Instrumento(configuraciones)

Referencias: Recolectar datos

Precondiciones: Es necesario que los instrumentos estén debidamente conectados.

Postcondiciones: Guarda en memoria una instancia de un objeto heredado de la clase instrumento.

Contrato código: loop()

Operación: loop()

Referencias: Recolectar datos

Precondiciones: Debió haber guardado las configuraciones.

Postcondiciones: Genera un ciclo “indeterminado”.

Contrato código: obtenerDatos

Operación: obtenerDatos()

Referencias: Recolectar datos

Precondiciones: Debió haber iniciado el instrumento.

Postcondiciones: Inicia la recolección de datos de un objeto de instrumento

Contrato código: recolectarDatos()

Operación: recolectarDatos()

Referencias: Recolectar datos

Precondiciones: Debió haber solicitado la función de obtener Datos

Postcondiciones: Guarda en memoria cada datos en forma de diccionario.

Contrato código: guardarDatos

Operación: guardarDatos()

Referencias: Recolectar datos

Precondiciones: Debió haber obtenido datos nuevos

Postcondiciones: Guarda los datos en memoria y deja como objeto “0” en memoria a “datos”

Contrato código: enviarDatosServidor

Operación: enviarDatosServidor()

Referencias: Recolectar datos

Precondiciones: Debe haber una base de datos en la raspberry

Postcondiciones: consulta y envía los datos guardados al servidor en formato json.

Recibir datos:

Contrato código: enviarDatosServidor

Operación: enviarDatosServidor()

Referencias: Recibir datos

Precondiciones: Debe estar en funcionamiento el programa.

Postcondiciones: recibe los datos en formato json y los guarda en memoria “datos”

Contrato código: buscarError

Operación: buscarError()

Responsabilidad: busca errores en los datos y si los hay envía correo y si no pasa.

Referencias: Recibir datos

Precondiciones: Debe estar en funcionamiento el programa y debe haber datos en memoria.

Postcondiciones:

Contrato código: guardarDatos

Operación: guardarDatos(datos)

Responsabilidad: guarda los datos en una base de datos.

Referencias: Recibir datos

Precondiciones: Debe haber en memoria los datos.

Postcondiciones:

Contrato código: generarGraficos

Operación: generarGraficos(datos)

Responsabilidad: genera datos recibidos, los gráficos, y los guarda en formato json

Referencias: Recibir datos

Precondiciones: Debe haber en la memoria datos.

Postcondiciones:

Administrador Nodos:

Sección añadir datos:

Contrato código: añadirNodo

Operación: añadirNodo()

Referencias: Administrar nodos

Precondiciones: Se debe ingresar como administrador

Postcondiciones: Crear nodo.

Contrato código: enviarDatos
Operación: enviarDatos(props)
Referencias: Administrar nodos
Precondiciones: Tener objeto nodo creado.
Postcondiciones: Se crea JSON con los datos.

Contrato código: peticionIdentificador
Operación: peticionIdentificador(props)
Referencias: Administrar nodos
Precondiciones: Se está conectado a la base de datos.
Postcondiciones: Se inserta un nuevo nodo a la base de datos.

Contrato código: nuevoldentificador
Operación: nuevoldentificador(props)
Referencias: Administrar nodos
Precondiciones: Se tiene creada la tabla de nodos..
Postcondiciones:

Sección editar datos:

Contrato código: editarNodo
Operación: editarNodo(id)
Referencias: Administrar nodos
Precondiciones: Se debe ingresar como administrador.
Postcondiciones:

Contrato código: enviarDatosEdicion
Operación: enviarDatosEdicion(props)
Referencias: Administrar nodos
Precondiciones: Confirmación de la existencia del nodo.
Postcondiciones: Se crea JSON con los datos.

Contrato código: editarDato
Operación: editarDato(props)
Referencias: Administrar nodos
Precondiciones: Se reciben los datos para editar en el estándar requerido para la función actualizarDato.
Postcondiciones: Se actualiza un nodo en la base de datos.

Contrato código: actualizarDato

Operación: actualizarDato(props)

Referencias: Administrar nodos

Precondiciones: Se tiene creado la tabla y el nodo requerido.

Postcondiciones:

Sección eliminar datos:

Contrato código: eliminarNodo

Operación: eliminarNodo(id)

Referencias: Administrar nodos

Precondiciones: Se ingresa como administrador

Postcondiciones:

Contrato código: confirmaEliminar

Operación: confirmaEliminar(id)

Referencias: Administrar nodos

Precondiciones: El nodo requerido está en la base de datos.

Postcondiciones: Se cambia el valor de advertencia a false.

Contrato código: eliminarNodo

Operación: eliminarNodo(id)

Referencias: Administrar nodos

Precondiciones: Se está conectado a la base de datos.

Postcondiciones: