## communicator SerialCommunicationIntefaceListener **{1}** <<interface>> SerialIntefaceObserver ser: Serial observers: List - listen\_to\_interface: Boolean + update(received\_message: String): void + attach(observer: SerialInterfaceObserver): void + detach(observer: SerialInterfaceObserver): void - notify(received\_message: String): void - read\_serial\_interface(): void <<interface>> + run(): void CommunicationInterfaceListener + stop(): void + attach(observer: SerialInterfaceObserver): void detach(observer: SerialInterfaceObserver): void

## SerialCommunicationIntefaceSender {1} - ser: Serial - write\_string(message\_string: String): void + send\_message(message: String): void + send\_message(message: String): void - communication interface sender(): CommunicationInterfaceSender + get\_communication\_interface\_listener(): CommunicationInterfaceListener - communication\_interface\_sender(): CommunicationInterfaceSender - communication\_interface\_sender(): CommunicationInterface\_sender(): Communication\_interface\_sender(): Comm

notify(received\_message: String): void

+ send\_message(message: String): void

+ *run*(): void