Proyecto Final Informática II:  
Documento de requisitos (en base a IEEE Std. 830-1998)

Estación Matera

# Autor

Lucas Ramírez

Universidad Tecnológica Nacional Facultad Regional San Francisco

lucaasraamirez@hotmail.com

# Resumen

Es máquina que requerirá de la intervención del usuario para programar la temperatura por medio de pulsadores y el rellenado del depósito de agua. El dispositivo va a calentar el agua a la temperatura deseada, una vez hecho dará aviso para empezar a usarlo.

# 1. Introducción

## Propósito

El propósito de este dispositivo es liberar al usuario de la tarea de cebar el mate, para poder concentrarse en su trabajo/estudio o por el simple hecho que no le guste cebar.

## Ámbito del sistema

* Se podrá dar un nombre al futuro sistema (Estación Matera)
* El dispositivo indicará un nivel de caga de agua, una temperatura de corte programada con en el encendido, detectará el mate para poder dispensar una cantidad pequeña de agua una vez que ya este caliente, Se explicará lo que el sistema hará y lo que no hará, de manera breve.
* En el futuro podría ser un sistema más compacto y más rápido para mayor comodidad y practicidad, también se podría hacer para que sea inalámbrico por medio de batería y podría ser programado por celular.

## Definiciones, Acrónimos y Abreviaturas

OPCIONAL: En esta subsección se mostrará una lista completa de todos los documentos referenciados en la ERS

# 2. Descripción general

## Descripción del producto

Realizar un diagrama en bloques que permita mostrar todas las partes del sistema. Acompañar de una explicación breve

## Restricciones

Esta subsección describirá aquellas limitaciones que se imponen sobre los desarrolladores del producto:

* Politicas de la empresa
* Limitaciones del hardware
* Interfaces con otras aplicaciones
* Operaciones paralelas
* Lenguaje(s) de programación
* Protocolos de comunicación
* Consideraciones acerca de la seguridad

## Requisitos futuros

OPCIONAL: Esta subsección esbozara futuras mejoras al sistema, que podrán analizarse e implementarse en un futuro.

# 3. Requisitos específicos

Esta sección contiene los requisitos a un nivel de detalle como para permitir a los diseñadores diseñar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas planificar y realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos. Todo requisito aquí especificado describirá comportamientos externos del sistema, perceptibles por parte de los usuarios, operadores y otros sistemas. Esta es la sección más larga e importante de la ERS. Deberán aplicarse los siguientes principios:

* El documento deberá ser perfectamente legible por personas de muy distintas formaciones e intereses.
* deberán referenciarse aquellos documentos relevantes que poseen alguna influencia sobre los requisitos.
* Todo requisito deberá ser unívocamente identificable mediante algún código o sistema de numeración adecuado.
* Lo ideal, aunque en la práctica no siempre realizable, es que los requisitos posean las siguientes características:
  + Corrección: La ERS es correcta si y solo si todo requisito que figura aquí (y que será implementado en el sistema) refleja alguna necesidad real. La corrección de la ERS implica que el sistema implementado será el sistema deseado.
  + No ambiguos: Cada requisito tiene una sola interpretación. Para eliminar la ambigüedad inherente a los requisitos expresados en lenguaje natural, se deberán utilizar gráficos o notaciones formales. En el caso de utilizar términos que, habitualmente, poseen más de una interpretación, se definirán con precisión en el glosario.
  + Completos: Todos los requisitos relevantes han sido incluidos en la ERS. Conviene incluir todas las posibles respuestas del sistema a los datos de entrada, tanto validos como no válidos.
  + Consistentes: Los requisitos no pueden ser contradictorios. Un conjunto de requisitos contradictorio no es implementable.
  + Clasificados: Normalmente, no todos los requisitos son igual de importantes. Los requisitos pueden clasificarse por importancia (esenciales, condicionales u opcionales) o por estabilidad (cambios que se espera que afecten al requisito). Esto sirve, ante todo, para no emplear excesivos recursos en implementar requisitos no esenciales.
  + Verificables: La ERS es verificable si y solo si todos sus requisitos son verificables. Un requisito es verificable (testeable) si existe un proceso finito y no costoso para demostrar que el sistema cumple con el requisito. Un requisito ambiguo no es, en general, verificable. Expresiones como a veces, bien, adecuado, etc. Introducen ambigüedad en los requisitos. Requisitos como “en caso de accidente la nube toxica no se extenderá mas allá de 25Km" no es verificable por el alto costo que conlleva.
  + Modificables: La ERS es modificable si y solo si se encuentra estructurada de forma que los cambios a los requisitos pueden realizarse de forma fácil, completa y consistente. La utilización de herramientas automáticas de gestión de requisitos (por ejemplo RequisitePro o Doors) facilitan enormemente esta tarea.
  + Trazables: La ERS es trazable si se conoce el origen de cada requisito y se facilita la referencia de cada requisito a los componentes del diseño y de la implementación. La trazabilidad hacia atrás indica el origen (documento, persona, etc.) de cada requisito. La trazabilidad hacia delante de un requisito R indica que componentes del sistema son los que realizan el requisito R.

## Requisitos de hardware e interfaces externas

Se describirán los requisitos que afecten a la interfaz de usuario, interfaz con otros sistemas (hardware y software), interfaces de comunicaciones y componentes de hardware a utilizar.

## Requisitos de software

Esta subsección (quizá la más larga del documento) deberá especificar

todas aquellas acciones (funciones) que deberá llevar a cabo el software. Normalmente (aunque no siempre), son aquellas acciones expresables como \el sistema deberá . . . ". Si se considera necesario, podrán utilizarse notaciones gráficas y tablas, pero siempre supeditadas al lenguaje natural, y no al revés.

Como nuestro proyecto generalmente tiene dos partes una en Arduino y otra en PC, vamos a dividir los requisitos de acuerdo a cada parte del sistema:

* Requisitos de firmware (Arduino):
  + RAR1: El arduino debe leer la señal del sensor…
* Requisitos de software de la interfaz de usuario (PC)
  + RIU1: La interfaz debe…
  + RIU2: …