



Programa de licenciatura en Ingeniería en Computadores
Proyecto de diseño de Ingeniería en Computadores
Laboratorio LuTec

Documento de requerimientos

Proyecto:

KROTIC Architecture Upgrade

Profesor:

Gustavo Cubas Euceda

Estudiante:

Marcelo Sánchez Solano

I Semestre, 2021

Tabla de Contenidos

Propósito	3
Alcance del proyecto	3
Resumen del producto	3
Perspectiva del producto	3
Interfaces del sistema	4
Interfaces de usuario	4
Interfaces de Hardware	4
Interfaces de software	5
Interfaces de comunicación	5
Restricciones de memoria	6
Operaciones	7
Adaptaciones del sitio	7
Interfaces con servicios	7
Funciones del producto	7
Características del usuario	9
Limitaciones de software	10
Limitaciones de hardware	10
Definiciones	10
Siglas y abreviaturas	11
Requisitos	11
Requisitos de interfaz	11
Funciones	11
Requisitos de usabilidad	14
Requisitos de rendimiento	14
Requisitos de la base de datos lógica	15
Restricciones de diseño	15
Atributos del sistema de software	15
Información de soporte	15
Verificación	15
Apéndices	16
Supuestos y dependencias	16
Matriz de Trazabilidad	16
Referencias	17

Introducción

Propósito

El propósito principal del proyecto consiste en actualizar la arquitectura que se tiene actualmente para el robot KROTIC. Se propone que la nueva arquitectura posea dos niveles de operación. Donde un nivel se dedique a controlar las funciones básicas del robot y otro se encargue de manejar funciones superiores, como establecimiento de protocolos o descomponer las tareas por realizar en la arquitectura.

Alcance del proyecto

Para el proyecto solo se abarca el diseño de arquitectura de conexión de los controladores para el robot. El diseño no se relaciona con otras partes en desarrollo del proyecto microSCADA, como el sistema de cámaras que graban al robot. El diseño que se debe proponer debe poseer al menos dos niveles de operación, los cuales se controlan con dispositivos tales como Raspberry Pi, Beaglebone Black, microcontroladores y otros elementos afines, que se encargan de manejar las funciones del robot. Además, el nuevo diseño debe soportar un módulo de comunicación WiFi ya que es requisito para la interacción efectiva con los estudiantes que utilizan el robot.

Resumen del producto

Perspectiva del producto

Aunque el diseño de la arquitectura forma parte del proyecto KROTIC microSCADA, no se relaciona con ninguna otra parte del proyecto, como el sistema de grabación o con el sistema de información. En este proyecto solo se toma en cuenta las funciones y la autonomía del robot. Incluye funciones de grabación y transmisión, pero en la cámara propia del robot.

La arquitectura debe ser propuesta de modo que se permita la flexibilidad de uso del robot. Esto implica que para los proyectos que desarrollan los estudiantes, debe ser capaz de utilizar solo alguna parte del robot o varias al mismo tiempo. Esto porque la idea es que los estudiantes propongan proyectos, en lugar de trabajar proyectos ya acordados. La idea es que los estudiantes propongan proyectos que puedan tener diferentes enfoques y que puedan usar la creatividad de los estudiantes.

Incluso en el caso de que algún proyecto propuesto necesite un módulo que no está disponible en el momento, el robot posee una extensión que utiliza placas de desarrollo *protoboard* para que el módulo necesario sea implementado. Entonces, lo

que se quiere tener es flexibilidad para poder manejar todo tipo de proyectos en el KROTIC.

Interfaces del sistema

Las interfaces con factores externos corresponden a los sensores de movimiento, luz, temperatura y sonido. Además, el robot posee una cámara propia que se desea que transmita en directo a los estudiantes.

Interfaces de usuario

Aunque el sistema no tiene una interfaz que interactúa con usuarios finales, se tendrá un banco de pruebas del sistema que evaluarán el desempeño y consumo al ejecutar las diferentes operaciones del robot. Estas pruebas se realizan a través de un software de simulación y por lo tanto requieren manejo de los componentes a través de la interfaz de la simulación.

Interfaces de Hardware

En general, las plataformas de hardware a ser utilizadas no están actualmente especificadas ya que se evaluarán diferentes opciones más adelante durante la etapa de diseño de la arquitectura. Se pueden mencionar alternativas que fueron utilizadas en iteraciones anteriores como:

1. **Beaglebone Black.** Una placa de desarrollo similar a Raspberry Pi. Que tiene un procesador con arquitectura ARM Cortex-A8, con 512 MB de memoria RAM y 2GB para la memoria secundaria. Además, la tarjeta tiene seis entradas analógicas y soporte para I²C. También, la placa Beaglebone Black Wireless trae soporta conexión WiFi implementada [\(1\)](#).
2. **Raspberry Pi.** El último modelo de Raspberry lanzado en el 2020 tiene un procesador ARM Cortex-A72 con 4GB de RAM. Posee soporte para WiFi sin necesidad de módulos extra. Una diferencia con el Beaglebone Black es que no posee entradas analógicas, por lo tanto, se necesitaría un ADC externo para soportar lecturas analógicas. También soporta la comunicación I²C [\(2\)](#).
3. **Atmega328p.** Es un micro controlador basado en RISC de 8 bits, que tiene 23 pines de entrada y salida, 2 kB SRAM interna y 1 kB EEPROM. Similar a las opciones anteriores tiene capacidad para comunicación I²C, entre otras [\(3\)](#).
4. **Arduino.** De forma general, esta placa tiene capacidad para seis entradas analógicas y catorce digitales. Las especificaciones de memoria son iguales a las del Atmega328p ya que está integrado en la placa [\(4\)](#).
5. **ESP8266.** Placa de desarrollo especial para aplicaciones de IoT, con memoria RAM de 64kB, un pin para entrada analógica y 16 para digitales. Soporte para comunicación I²C, entre otras. Al ser utilizada mayoritariamente para IoT, el módulo de WiFi viene incluido [\(6\)](#).

El nuevo diseño, puede o no utilizar estas tarjetas de desarrollo, en el proceso de investigación y diseño se debe investigar por más alternativas y contrastar las propuestas para seleccionar la más adecuada para el proyecto.

Con respecto a los sensores que se usarán, son construidos en el mismo laboratorio, por lo que no se cuentan con detalles técnicos.

Finalmente, para el movimiento del robot se deberán utilizar servos Parallax, y en el caso de temporizadores se pueden utilizar los incluidos en las tarjetas de desarrollo o unos externos.

Interfaces de software

Para este proyecto, el uso de bibliotecas computacionales que se podría presentar, sería en la etapa de transmisión de datos a través de WiFi. La implementación estará sujeta a las evaluaciones que se hagan a las diferentes opciones encontradas, que se tomarán más adelante.

Interfaces de comunicación

La comunicación entre los componentes de procesamiento se realizará a través del protocolo I²C, ya que cuando se haga la selección de diseño de arquitectura, es muy probable que los dispositivos no sean del mismo fabricante.

El protocolo de comunicación integrada o I²C, combina partes de protocolos como SPI y UART. Específicamente, se pueden conectar múltiples dispositivos esclavos a un solo maestro [\(7\)](#), como en SPI o viceversa; múltiples maestros controlando un solo esclavo. Se asemeja a UART ya que solo se necesitan dos líneas de comunicación entre los dispositivos, como se ejemplifica en la figura 1.

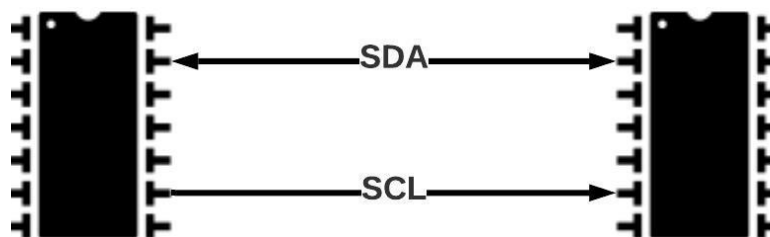


Figura 1. Puertos de comunicación para el protocolo I²C.

Para la comunicación desde el laboratorio con los estudiantes, en los casos que se utilice KROTIC de forma remota, se utiliza la comunicación por internet usando WiFi. Por lo que es necesario que los diseños que se propongan soporten esta comunicación. La comunicación WiFi se usa para la transmisión de valores leídos por los sensores, para la transmisión de imágenes y videos con los

estudiantes. El diagrama de la figura 2 muestra las interacciones de las partes con su respectivo protocolo de comunicación.

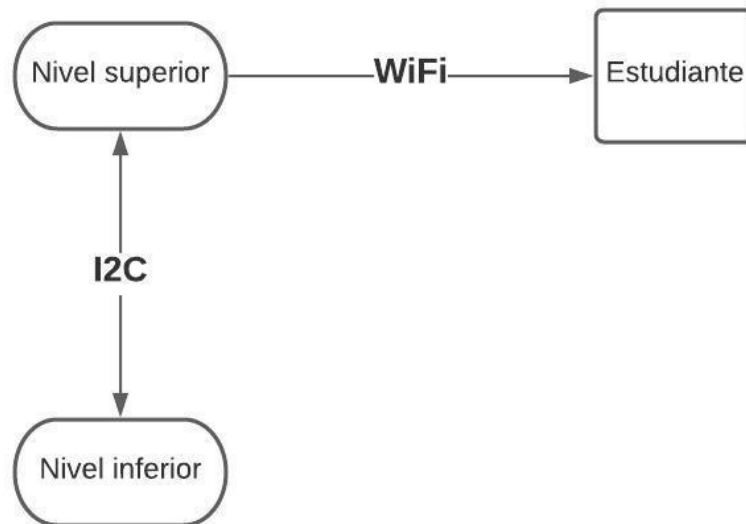


Figura 2. Diagrama de protocolos inicial de comunicación entre las partes del sistema.

Restricciones de memoria

La memoria estará definida por la memoria disponible en los dispositivos de procesamiento de cada nivel de la arquitectura. Los cuales están por definir durante la etapa de diseño.

La restricción que actualmente existe es que sea equivalente o superior a la configuración que se tenía en la iteración anterior, que consiste en la memoria de un Arduino y un Beaglebone Black. En ese caso, la memoria que se tiene como base es poco más de 512 MB. La memoria no es necesario que tenga gran capacidad ya que los datos que se almacenan son valores de sensores, que no constituyen mayor ocupación. Los datos que más memoria necesitan son: el programa, imágenes y videos captados directamente por el robot. Como se quiere que estos datos sean enviados por WiFi no deberían ocupar gran cantidad de memoria por mucho tiempo.

Operaciones

Dado que se está hablando del diseño de una arquitectura, se pueden mencionar las acciones que el robot KROTIC realiza cuando se ejecuta un programa. Estas son:

1. Avanzar una cierta distancia hacia adelante.
2. Retroceder una cierta distancia.
3. Girar sobre su propio eje, tanto a la izquierda como a la derecha.
4. Utilizar los sensores:
 - a. Luz.
 - b. Sonido.
 - c. Movimiento.
 - d. Temperatura.
5. Apagar y encender luces.
6. Temporizar acciones.
7. Capturar a través de la cámara del robot las acciones y su entorno.

Como parte de las operaciones que realiza el robot, se involucra el manejo de comportamientos que pongan en peligro algún componente de la arquitectura, por lo que, en caso de ocurrir, el robot suspende operaciones.

Finalmente, el robot también puede tomar video o fotografías durante la ejecución del programa, que son enviadas a los estudiantes encargados del proyecto.

Adaptaciones del sitio

El robot está destinado para que pueda ser armado por el asistente de laboratorio, que se asume es alguien con conocimiento previo sobre el robot y su funcionamiento. Sin embargo, también se quiere que cualquier persona interesada pueda ser capaz de armar el robot dados los respectivos diagramas y documentación respectiva.

Interfaces con servicios

No aplica.

Funciones del producto

En general, el sistema se divide en dos niveles: en el nivel superior de operación que controla la cámara propia del robot, el segundo nivel de operación que controla las operaciones básicas del robot. Entonces la división de funciones de cada nivel se resume de la siguiente manera:

1. Nivel superior.
 - a. Capturar fotos y videos durante la ejecución de un programa.
 - b. Transferir imágenes y videos capturados.

- c. Transferir valores de sensores, medidos por el nivel inferior.
 - d. Recibir los datos del nivel inferior.
 - e. Planificación y división de tareas para llevar a cabo las instrucciones dadas por el programa del estudiante. Incluye evaluación de las distintas opciones para realizar la misma tarea basado en consumo y complejidad.
2. Nivel inferior.
- a. Realizar mediciones con los distintos sensores.
 - b. Temporizar las acciones del robot.
 - c. Controlar el movimiento del robot.
 - d. Protección del robot.

Las responsabilidades anteriores se resumen en el siguiente árbol de funcionalidades y en el diagrama general de ambos niveles:

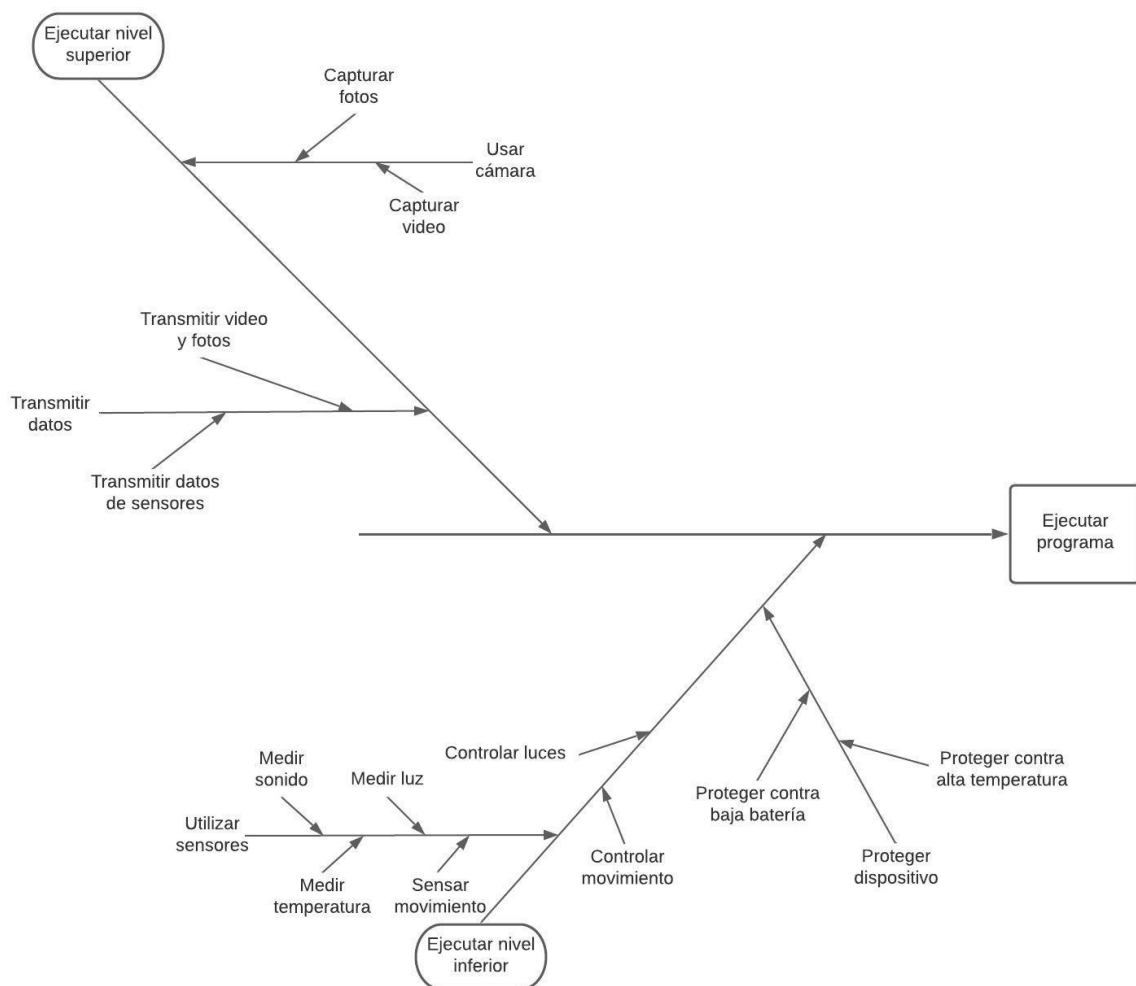


Figura 3. Árbol de funcionalidades de la arquitectura.

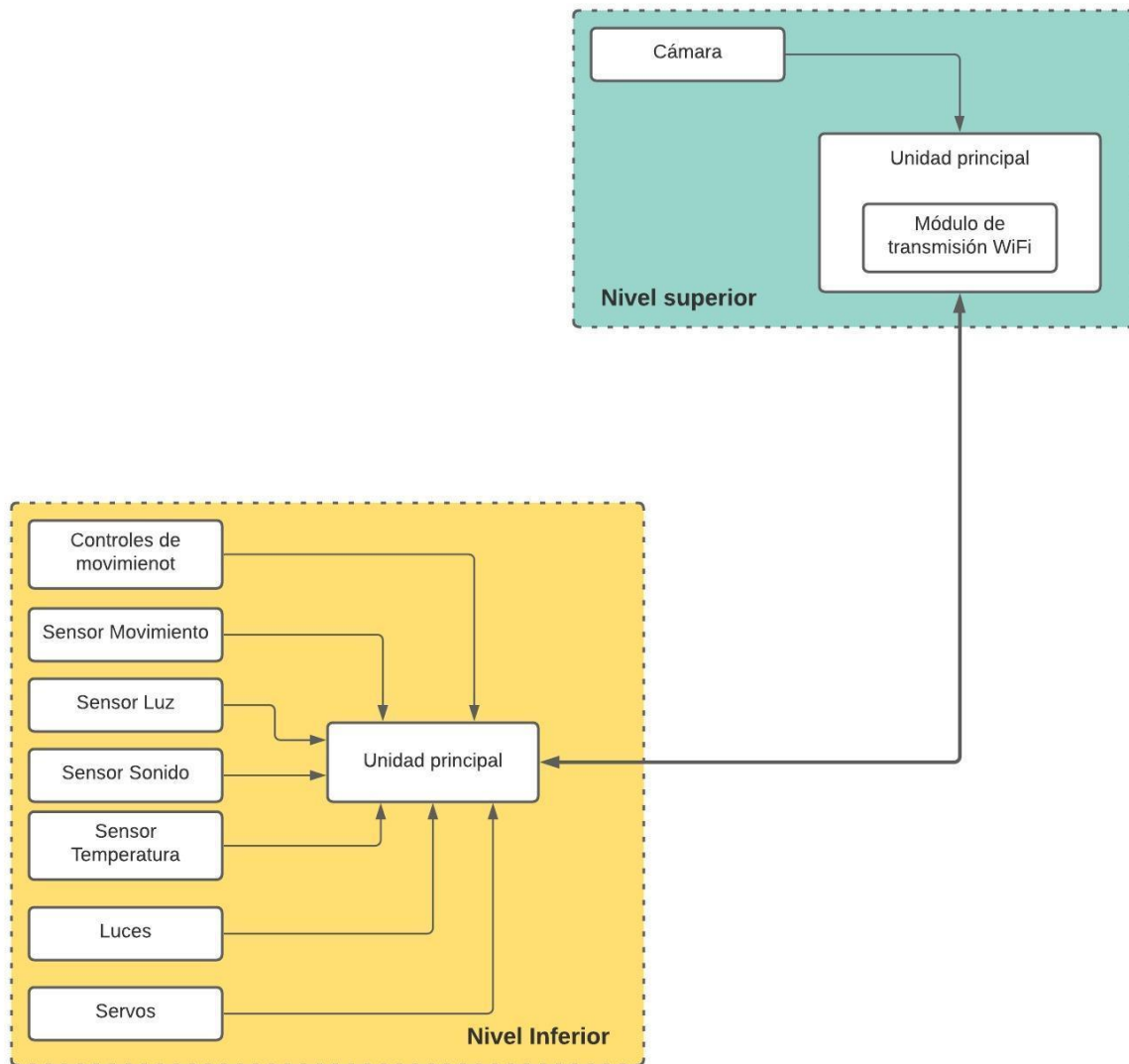


Figura 4. Diagrama general de los componentes de cada nivel.

Características del usuario

Al ser una plataforma de bajo nivel, no estará interactuando con un usuario final. Sin embargo, la persona que sea responsable de mantener la arquitectura, en este caso, el asistente del laboratorio, debe conocer las especificaciones del diseño y la programación de los dispositivos. Por lo tanto, se prefiere que tenga conocimientos de un estudiante de ingeniería en computadores; es decir, con conocimientos tanto en hardware como en software y no está limitado a estudiantes tampoco. Se extiende a que cualquier persona pueda interactuar con el diseño, dadas las herramientas necesarias para hacerlo, como diagramas y documentación.

Limitaciones de *software*

Las limitaciones de software no se tienen especificadas en este momento, ya que dependerá de las necesidades de software que tengan las plataformas de hardware seleccionadas para los diseños.

Limitaciones de *hardware*

Al ser un robot un cubo, las partes que serán parte del diseño deben caber dentro del robot que se tiene actualmente. De igual forma, se debe contemplar el peso de los componentes ya que esto podría limitar las capacidades de movimiento del robot.

Otro aspecto a considerar, son los datos que se transmiten por WiFi, dado que los estudiante meta del proyecto general, en la mayoría de las ocasiones se ubican en lugares donde el acceso a internet es reducido; la transmisión debe ser la mínima posible.

Luego, aunque se tiene libertad de agregar módulos en los diseños que se van a proponer más adelante, se tiene que tomar en cuenta el consumo de energía total, se espera que no sea muy alto, ya que esto influye en las baterías a ser utilizadas, cuyo valor exacto será calculado una vez se hayan diseñado las propuestas.

Definiciones

1. **KROTIC.** Un kit de robótica de bajo costo, implementado por el laboratorio LuTec de forma modular, donde cada parte del kit puede ser usado independientemente. De igual forma, los módulos pueden ser agregados y removidos fácilmente según sea posible.
2. **microSCADA.** Acrónimo para *Supervisory Control And Data Acquisition*, concepto empleado para el control y supervisión de software y hardware a distancia. Facilita retroalimentación en tiempo real con dispositivos y control de procesos automáticos.
3. **Microcontrolador.** Se refiere a una computadora de menor tamaño, ensamblada en un solo circuito integrado. El integrado contiene al menos una unidad de procesamiento, memoria, pines de entrada y salida. Los microcontroladores son normalmente utilizados para aplicaciones empotradas.
4. **WiFi.** Tecnología de transmisión inalámbrica.
5. **Arquitectura.** Diseño de interconexión de los dispositivos que componen un sistema.
6. **Protoboard.** Placa de desarrollo experimental, donde se pueden insertar componentes electrónicos y cables para armar circuitos sin la necesidad de soldar ningún componente.
7. **SPI.** Estándar de comunicaciones para comunicar distintos circuitos integrados.

8. **UART.** Protocolo de transmisión configurable de datos seriales asíncronos.
9. **Dispositivo maestro.** Dispositivo encargado de controlar el funcionamiento de otros dispositivos. Funciona como el centro de comunicaciones, en caso de tener muchos dispositivos.
10. **Dispositivo esclavo.** Dispositivo que es controlado por otro dispositivo maestro.

Siglas y abreviaturas

1. **SPI.** Serial Peripheral Interface (Interfaz periférica serial).
2. **UART.** Universal Asynchronous Receiver-Transmitter (Emisor-receptor universal asíncrono).
3. **KROTIC.** Kit de Robótica Costarricense

Requisitos

Requisitos de interfaz

No aplica.

Funciones

Para esta sección se utiliza una tabla para cada uno de los requerimientos identificados, mostrando su identificador único. La sintaxis utilizada para redactar los requerimientos sigue el formato EARS.

Identificador	Nombre	Descripción
KR-01	Niveles de arquitectura.	La arquitectura deberá tener al menos dos niveles de operación.
KR-02	Control de los niveles de la arquitectura.	Cada nivel será controlado por un microcontrolador, microprocesador o micro computadora. Según se considere adecuado.
KR-03	Comunicación entre dispositivos.	La comunicación entre los dispositivos de control de cada nivel se debe hacer a través del

		protocolo I ² C.
KR-04	Uso de cámara propia del robot.	El nivel superior deberá procesar los datos de la cámara propia del robot.
KR-05	Funciones de la cámara del robot.	La cámara del robot deberá ser capaz de tomar fotos y videos de su entorno.
KR-06	Transmisión de datos de cámara.	Las fotos y videos tomados desde la cámara propia del robot deberán ser transmitidos en tiempo real.
KR-07	Sensor de luz.	El nivel inferior deberá manejar tres sensores de luz.
KR-08	Sensor de movimiento.	El nivel inferior deberá manejar un sensor de movimiento.
KR-09	Sensor de temperatura.	El nivel inferior deberá manejar un sensor de temperatura.
KR-10	Sensor de sonido.	El nivel inferior deberá manejar al menos un sensor de sonido.
KR-11	Controlar el movimiento del robot.	El nivel inferior deberá controlar el movimiento del robot usando dos servos.
KR-12	Transmisión de datos de sensores.	El nivel superior debe transmitir los datos leídos a través de los sensores del nivel inferior.

KR-13	Movimiento hacia adelante del robot.	El robot debe poder avanzar hacia adelante treinta centímetros (30 cm).
KR-14	Movimiento en reversa del robot.	El robot debe poder avanzar hacia atrás treinta centímetros (30 cm).
KR-15	Movimientos giratorios del robot.	El robot debe poder girar sobre su propio eje una cierta cantidad de grados (múltiplos de 30°).
KR-16	Encender luces del robot.	Las luces se encienden si el programa del estudiante así lo requiere.
KR-17	Apagar luces del robot.	Las luces se apagan si el programa del estudiante así lo requiere.
KR-18	Temporizar actividades.	El temporizador se utilizará si el programa del estudiante así lo requiere.
KR-19	Programa de ejecución.	El nivel superior será el encargado de recibir el programa creado por los estudiantes.
KR-20	División de tareas del programa.	Fuera de alcance.
KR-21	Protección en caso de baja batería.	Si se detecta batería baja en alguno de los dispositivos principales, se suspenderán las acciones inmediatamente.
KR-22	Protección en caso de alta temperatura.	Si se detecta alta temperatura en alguno de los dispositivos de control, se suspenderán

		las acciones inmediatamente.
--	--	------------------------------

Requisitos de usabilidad

Ya que no se tiene una interfaz de usuario final, excepto la mencionada anteriormente en la sección de [interfaces de usuario](#), por lo que las características de usabilidad del banco de simulaciones no depende del sistema desarrollado, sino por el software de simulación que se utilice.

Requisitos de rendimiento

Identificador	Nombre	Descripción
KR-23	Entradas soportadas.	El diseño deberá soportar la cámara del robot, un sensor de temperatura, tres sensores de luz, un sensor de movimiento, un sensor de sonido, un temporizador y dos servos.
KR-24	Temperatura de operación.	La temperatura de los dispositivos empleados no deberá sobrepasar los rangos normales de un componente eléctrico. El rango normal se ubica en el intervalo de los 0 °C y los 70 °C.
KR-25	Captura de videos.	Los videos tomados por la cámara del robot no deben sobrepasar cinco minutos de duración (5 min).
KR-26	Transmisión de videos.	Para un video de la duración mencionada, el tiempo de transmisión no debe exceder los seis minutos (6 min).

KR-27	Transmisión de fotos.	La duración de transmisión de las fotos tomadas por el robot no debe exceder un minuto (1 min).
-------	-----------------------	---

Requisitos de la base de datos lógica

La arquitectura no incluye una base de datos.

Restricciones de diseño

En este caso, el robot se ubica en el laboratorio LuTec, por lo tanto, aspectos tales como dimensiones y espacio disponible son dictados por dicho laboratorio. Otro aspecto que se debe tomar en cuenta a la hora de tomar decisiones de diseño es el factor ambiente, se quiere que la arquitectura sea amigable con el ambiente en cuanto al consumo de energía principalmente.

Atributos del sistema de *software*

1. **Confiable.** La arquitectura debe soportar todas las entradas necesarias (sensores, cámaras, entre otros) sin que las unidades de procesamiento fallen. Las comunicaciones tanto entre los niveles de operación como la comunicación con el estudiante debe estar funcionando los seis minutos que dura la interacción.
2. **Disponibilidad.** No aplica.
3. **Seguridad.** No aplica.
4. **Mantenibilidad.** No aplica.

Información de soporte

El programa que recibe la arquitectura se encuentra en un lenguaje de programación propio que cuenta con su propio intérprete. Por lo tanto, el nivel superior debe manejar este intérprete.

Verificación

Muchas de las verificaciones que se harán se concentrarán en simulaciones utilizando algún programa de software. Sin embargo, estas no están especificadas explícitamente ya que dependerá del diseño que se elija en fases posteriores del proyecto. De igual forma, a manera general se pueden mencionar algunas pruebas básicas que se deberán realizar:

1. **Captura de información de sensores.** Esta prueba consiste en verificar la lectura correcta de los sensores de la arquitectura.
2. **Comunicación de dispositivos.** Se observará la transmisión de datos entre los dispositivos de los niveles de la arquitectura para verificar su correcto envío y recepción.
3. **Lectura y almacenamiento del programa.** Consiste en verificar la correcta lectura de los programas que se ejecutan en la arquitectura.
4. **Transmisión de datos.** Consiste en obtener tiempos de transmisión de los distintos tipos de datos que se manejan. Además de verificar la recepción correcta.
5. **Manejo de todos los dispositivos.** Se verificará que las plataformas de desarrollo que se vayan a seleccionar puedan soportar el manejo de todos los dispositivos que se necesitan.

Apéndices

Supuestos y dependencias

Se supone que el laboratorio provee los componentes utilizados en iteraciones anteriores y que se tienen que utilizar en este diseño, como los sensores y servos. También, se asume que, en caso de necesitarlo, nuevos componentes serán obtenidos a través del laboratorio.

Matriz de Trazabilidad

No aplica.

Referencias

- [1] BeagleBone black Wireless. Tomado de: <https://beagleboard.org/black-wireless>

- [2] Raspberry Pi. Raspberry Pi 4 Model b specifications. Tomado de: <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/>

- [3] Atmega328p. Tomado de: <https://www.microchip.com/wwwproducts/en/ATmega328P>

- [4] Arduino Uno Pin Diagram, Specifications, Pin Configuration & Programming. Tomado de: <https://components101.com/microcontrollers/arduino-uno>

- [5] NodeMCU ESP8266 Pinout, Specifications, Features & Datasheet. Tomado de: <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet>

- [6] Valdez, J., & Becker, J. (2015). *Understanding the I2C Bus* (Rep.). Tomado de: https://www.ti.com/lit/an/slua704/slua704.pdf?ts=1614746377385&ref_url=https%253A%252F%252Fwww.google.com%252F