

# Reporte reto: “Robot Guardameta”

## Implementación de sistemas mecatrónicos (Grupo 301)

Jonatan David De La Rosa Patlan<sup>1</sup>, Andrea Michell Sarmiento García<sup>2</sup>, Leonardo Elioenait Galán Cruz<sup>3</sup>, Miguel Ángel Barragán Cruz<sup>4</sup>, Erick Thomas Carranza Perez<sup>5</sup>, David Torres Everest<sup>6</sup>

**Resumen—** Este documento electrónico contiene toda la información sobre el antes, durante y después de las áreas de electrónica, mecánica, control y organización de proyectos de la realización del reto “Robot guardameta”.

**Palabras clave—** electrónica, mecanismos, robot, lineal, programación, velocidad, organización.

### I. INTRODUCCIÓN

Actualmente la mecatrónica es una parte fundamental de nuestra vida cotidiana pues nos permite realizar diversas actividades de manera accesible y fácil. Es por ello que como estudiantes en este proyecto nos retamos a nosotros mismos a realizar un robot donde se integren las 4 áreas de la misma.

El reto consiste en la realización de un robot que realiza un movimiento lineal para cubrir una “portería” además de que mediante un mecanismo puede pegarle a una pelota.

Para ello es importante conocer diversos temas que durante 5 semanas vimos en clase de los cuatro módulos diferentes. También es vital la integración de componentes mecánicos, electrónicos y de programación.

### II. PROTOTIPADO

#### A. Diseño del prototipo

El diseño del prototipo del robot fue realizado en la primera clase. Decidimos realizarlo de forma circular dividido en dos partes: una donde se encontraría el solenoide y la otra que pegaría con la pelota. También decidimos darle un toque único y hacerlo característico con el logo de Batman en la parte superior como se muestra en la *figura 1* y a partir de ahí decidimos que nuestro proyecto se tornaría en base a esa temática. Finalmente, en el prototipo incluimos una tapa en la parte de arriba para los componentes electrónicos; sin embargo, en

la entrega final nos dimos cuenta de que no era necesario y terminó descartada.



Figura 1. Prototipo de robot

En cuanto al diseño del prototipo de la caja fue un proceso diferente pues hubo varios cambios en el mismo. Primero, realizamos el diseño con dos agujeros en la parte de la tapa empezando por el agujero del robot y después el agujero donde cae la pelota; sin embargo, nos dimos cuenta de que la pelota se trabaría con el agujero del robot y nunca llegaría al agujero de atrás (el de la pelota) como se muestra en la *figura 2*.

Por esa razón decidimos intercambiar el orden para mejor acceso a la pelota. Gracias a este prototipo, después agregamos más detalles como una rampa, el logo de Batman, agregamos otro agujero etc.

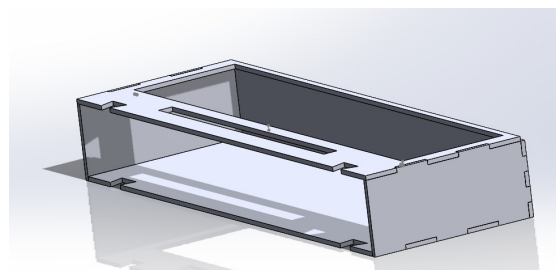


Figura 2. Primera caja diseñada

### B. 3D o MDF

Debido a los objetivos que le asignamos a cada pieza decidimos que tipo de material usar, aquí presentamos las razones:

#### CAJA: MDF

- Tiene dureza y rigidez
- Es más resistente y no se deforma fácilmente.
- Puedes tener cortes más precisos lo cual hace las dimensiones y formas perfectas.
- Tiene superficie lisa y uniforme lo cuál sirve para figuras básicas como rectángulos, cuadrados, círculos etc.
- Incluye construcción.

#### ROBOT Y SOPORTES: 3D

- Es muy versátil trabajar con él ya que puedes realizar casi cualquier cosa que te propongas.
- Lo podemos personalizar a como nosotros queramos, con la forma y diseño cualquiera.
- El material puede llegar a ser muy resistente.
- No necesita mano extra

### C. Diseño y costos

Después de realizar una investigación acerca de las diversas maneras de crear movimientos lineales decidimos que la opción más viable para llegar al objetivo final del reto era mediante un perfil en V que contiene un “carrito” con una banda dentada y poleas que ayudan al movimiento. A partir de ello, decidimos el diseño de todo lo demás, pues son dependientes del mismo. Tanto de la caja como del robot, por ejemplo, hablando del cuello del robot el cual tiene que ir montado al carrito fue hecho a la medida del mismo.

A pesar de que ya teníamos establecido el diseño, hubo varias fallas que conforme nos dábamos cuenta de los posibles errores, fuimos cambiando tal como se muestra en el apartado VII de anexos donde se muestran todos los documentos de los re-trabajos realizados donde se muestran los cambios conforme al paso del tiempo. Debido a esa causa también se aumentaron los costos, tal como se muestra en la *tabla 1* pues la necesidad de más material aumentó.

Materiales	Costos
Placas de MDF	\$360
PLA	\$500
Componentes electrónicos y mecánicos	\$1550
Impresiones	Gratis
Cortes de MDF	\$280
FINAL	\$2690

Tabla 1. Tabla de materiales

### D. Mejoras

Como todo proyecto, siempre existen mejoras a realizar para un mejor resultado.

En nuestro caso, podemos ver como área de oportunidad la posibilidad de realizar un mejor diseño interno de la caja, esto para mejor acomodo de la parte electrónica ya que cuando el robot se puso en marcha los cables por dentro estaban muy desacomodados lo que causaba que se desconectan o que sufrieran daños. Por lo cuál proponemos como mejora dejar un espacio determinado dentro de la caja solo para la electrónica con cables más resistentes y largos.

Por otra parte, una mejora más es la potencia y velocidad del solenoide, el cuál puede mejorarse con una fuente de energía externa la cual le proporcione los 24V que necesita y soporta.

### III. ELEMENTOS MECÁNICOS

#### A. Materiales y Fabricación

En la *tabla 2* se presenta el BOM utilizado en este proyecto.

Material	Num-	Costo	Función
Banda dentada	1	\$120	Transmitir el movimiento y la potencia entre polea y motor
Motor	1	\$292	Generar movimiento y potencia para impulsar al carrito
Polea motor	1	\$120	Proporciona una superficie de agarre para la banda dentada y transferir el movimiento giratorio del motor al carrito
Polea fija	1	\$140	Guiar y sostener la banda dentada en su lugar
Carro deslizante	1	\$294	Desplazar el robot
Driver	2	\$70	Suministrar la energía y controlar la velocidad y dirección del motor
Solenoides	1	\$70	Generar movimiento para pegar a la pelota
Módulo Bluetooth	1	\$176	Comunicación entre la aplicación y el robot
Limit Switches	2	\$50	Limitar el desplazamiento del robot

Cables	20	\$100	Proporcionar una conexión eléctrica confiable y segura.
Protoboard	1	-	Lugar para concentrar todos los componentes necesarios
Fuente de energía	1	-	Proporcionar voltaje necesario para hacer funcionar los componentes
Pegamento 5000	1	\$30	Para unir las piezas de la caja de MDF
Perfil en V	1	\$130	Tramo por el cual se desplaza el carrito deslizante
Placa de MDF	2	\$360	Material utilizado para cortar el diseño de la caja
PLA	1	\$500	Material para impresiones 3D de los soportes y el robot
Tornillo M3	4	\$2	Unir motor con soporte
Tornillos M5	2	\$2	Unir tapa del portero
Tuercas en T con tornillo	6	\$5	Unir el soporte que va en la caja al perfil V
Tuercas M5	2	\$2	Sostener el tornillo con la polea y el carrito al cuello
Tornillo M7	1	\$1	Unir carrito deslizante a cuello del robot
Pijas de madera m5	8	-	Unir los soportes a la

½ pulgada			caja
Cinchos	2	\$45	Tensar la banda

Tabla 2. Tabla de todos los materiales

### B. Actividades de diseño

Requerimientos especificados para la caja y portero:

- Caja: 70cm x 20cm x 10 cm
- Robot: 7cm x 6cm x 5cm

1. En base a las especificaciones que se nos otorgaron diseñamos el primer prototipo de caja el cual contaba con el espacio para que recorriera el portero y en la parte de atrás una rampa para la recolección de la pelota en caso de gol. Observar *figura 3*. (INSERTAR IMAGEN VECTORIAL)
2. Comentandolo entre el equipo vimos que el hecho de que la rampa estuviera en la parte de atrás podría afectar a que nunca se llegara a utilizar debido a la fuerza con la que golpeaban los solenoides por lo cual modificamos la caja haciendo una pequeña rampa en la parte de adelante y modificando el ancho de la caja ya que no cumpliera con los 20cm especificados. Observar *figura 4*, (INSERTAR IMAGEN VECTORIAL)
3. Una vez teniendo la caja fuimos al tablero para empezar a diseñar el cuello midiendo cómo había quedado nuestra caja con respecto al tablero original. Observar *figura 5*. (INSERTAR IMAGEN VECTORIAL)
4. Teniendo el cuello con la altura necesaria para librar los 8mm para la fricción a partir del tablero se diseñó un robot en el cual se le hizo una base para que el solenoide fuera metido a presión dentro de esta y con sus respectiva tapa. Observar *figura 6*. (INSERTAR IMAGEN VECTORIAL)
5. Con el robot impreso tomamos las medidas para el diseño de la pieza que iba a golpear la pelota tomando en cuenta que esta iba a ir a presión con la tuerca del solenoide y que necesitábamos que tuviera este hueco a la altura exacta a la que quedó la tuerca. Observar *figura 7*. (INSERTAR IMAGEN VECTORIAL)

### C. Explosiones Diseños Cad

En la *figura 9* observamos la vista explosionada de todo el ensamble con todas las partes utilizadas en el proyecto.

### D. Análisis de Velocidades

#### EXPERIMENTAL

- Tiempo tomado con cronómetro: 1.6s
- Distancia del perfil: 40cm

Velocidad:  $d/t$

$$\text{Velocidad: } 0.4m/1.6s = 0.25m/s$$

#### TEÓRICA

- Velocidad del motor: 200 RPM
- Diámetro de paso de polea de entrada (del motor): 5mm
- Dientes de polea fija: 20
- $w$  del motor= 2000 RPM
- Paso diametral de la banda dentada: 2mm  
= paso diametral de la polea

$$20 \text{ dientes} = 1 \text{ rev}$$

$$\text{distancia} = 1 \text{ rev} * \text{paso diametral} = 20 * 2 = 40mm$$

$$w = (2000 \text{ rev/min}) = (200 \text{ rev/6 s})$$

$$\text{Velocidad} = \text{diámetro de paso} / (\text{distancia} * w)$$

$$\text{Velocidad} = 5mm / (40mm * (200 \text{ rev} / 6 s))$$

$$\text{Velocidad} = 0.37 m/s$$

## IV. CONTROL

### A. Tabla E/S

Los pines son los siguientes mostrados en la tabla . Son en base al arduino Nano Atmega328P

Actuadores/ sensores	E/S	PIN Arduino	Variable
Solenoide	S. Digital	D12	Solenoide
Limit Switch 1	E.digit al	D2	LimitL
Limit Switch 2	E.digit al	D3	LimitR
A4988(Step)	S. Digital	D4	stepPin
A4988(Dir)	S. Digital	D8	dirPin

A4988(Enable)	S. Digital	D11	Enable
HC-05(TXD)	S. Analógica	D5 (PWM)	constructor : miBT
HC-05(RXD)	E. Analógica	D9 (PWM)	

Tabla 3. Entradas y salidas

### B. Arquitectura de control

La arquitectura de control utilizada fue la AVR RISC donde se utilizó el microcontrolador ATmega328p de un Arduino Nano, este fue seleccionado ya que esta arquitectura se caracteriza por su eficiencia en rendimiento, bajo consumo de energía y una facilidad de programación siendo el ATmega328P un microcontrolador de 8 bits basado en esta arquitectura AVR de Atmel.

#### Características del Microcontrolador:

- Velocidad de reloj de 16Mhz
- Memoria Flash 32 Kb
- Memoria Ram 2 Kb
- 23 pines E/S
- Comunicación UART, I2C y SPI
- ADC de 10 bits
- Oscilador interno de 8 bits

En el control se utilizaron diferentes componentes encargados de procesar información en específico para realizar diferentes tareas como el driver A4988 encargado del control del NEMA 17 y el módulo Bluetooth HC-05 encargado de la comunicación de control-robot

Además se hizo el desarrollo de una aplicación como control teniendo comunicación con el microcontrolador por medio de bluetooth mandando diferentes señales a este dependiendo de lo que se presionaba en la aplicación

### C. Algoritmos de Control

Los algoritmo de control se dividió en 3 bloques, siendo el desarrollo de la una aplicación, la comunicación Aplicación-Arduino y la comunicación Arduino-Robot

#### Desarrollo de app

Para la aplicación se hizo uso de un entorno de desarrollo visual y programación bastante sencillo "App Inventor" siendo un entorno bastante

amigable para el desarrollo de una aplicación sencilla donde se utiliza programación de bloques

Primeramente se hizo el desarrollo de Front End de la aplicación *figura 9* que cuenta con botones para dar dirección al robot así como otro boton en forma de círculo para accionar el solenoide

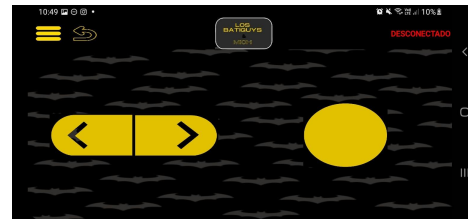


Figura 9. Front-End de la aplicación

También se hizo el desarrollo de un menú lateral *figura 10* en donde contamos con botones para realizar la conexión con el módulo HC-05 contando en la parte superior derecha un label de estado que puede decir el status de "DESCONECTADO" o "CONECTADO"

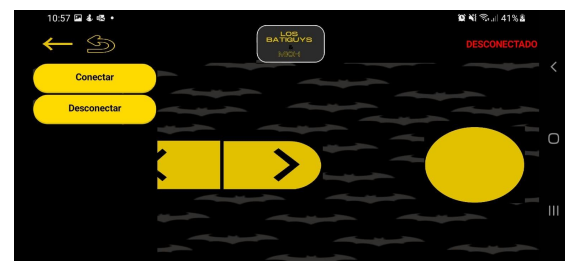


Figura 10. Menú lateral

#### Comunicación Aplicación-Arduino

La programación utilizada es bastante sencilla siendo de bloques esta, en donde primeramente tenemos que conectarnos por medio de los botones de conectar para establecer conexión, una vez establecida la conexión dependiendo de los botones que accionemos la aplicación llamará al Bluetooth client para mandar un letra, se se presiona izquierda mandara "L" si se presiona derecha "R" y al dejar de presionar cualquiera mandará "A" esto para tener el status de si se dejó de presionar o no, al mismo modo al presionar al círculo mandara "P" y al dejar de presionar mandara "S" como se puede ver en la *figura 11*.

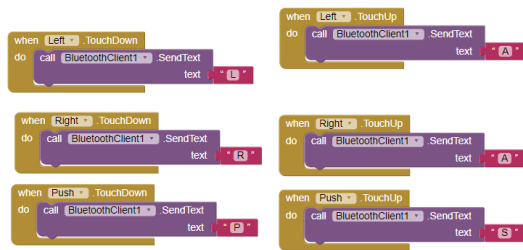


Figura 11. Menú lateral

### Comunicacion Arduino-robot

El desarrollo de la programación se realizó en C++ en Visual Studio Code por medio de PlataformaIO, este nos permite crear un ambiente de desarrollo para programar arduino teniendo que importar arduino como `#INCLUDE <Arduino.h>`, al igual para la conexión bluetooth se tuvo que hacer uso de `#INCLUDE <SoftwareSerial.h>` solo para la conexión bluetooth por medio de un constructor llamado “miBT”.

Como etapa inicial se tenía desconocimiento de algunos requerimientos de la programación como lo es el interrupt por lo que se programó una clase muy sencilla para los limit switches hecha por nosotros mismos llamada “limitS\_h” [Código en anexos] en donde meramente el constructor Limit s para declarar el pinMode como INPUT\_PULLUP así como un método que me devolvió el valor del límite, lo que facilitó al tener dos limit Switches el poder declararlos en una sola línea pero el tener que usar interrupts hizo que el tener esto no tenga mucho sentido pero se conservó.

Por la parte del código principal primeramente se hizo la declaración de todo los pines de salida y entrada así como los constructores, y las variables y en el void loop se realizó primero una comprobación por medio de condicional if si tenía lectura del sensor bluetooth de no ser así apagaba el motor, al tener lectura del sensor tenemos que la letra L en ASCII es 72, la R es 82, la A es 65 y la P 80 por lo que con esto, condicionales y pines digital podíamos controlar la dirección de giro del motor y el solenoide

Por último se agregó dos interrupts de modo CHANGE que hacían una especie de switch para apagar los motore y no poderse mover a ese lado del limt si seguía presionado

### *D. Sensores*

Se utilizaron 2 limit switches, uno colocado en la izquierda del perfil y otro en la parte derecha, para limitar el movimiento del portero y restringiendo la dirección de movimiento del motor. El limit switch izquierdo fue conectado al pin digital #2 del arduino, mientras que el limit switch derecho se conectado al pin digital #3, ambos switches mandan una señal digital constante de un 1 o alto, y al activarse el interruptor envían un 0 digital o bajo, a la vez que se enciende un led que indica que están activados. Los interruptores son alimentados por +3.3VDC proporcionados por el arduino. Es importante mencionar que ambos sensores cuentan con resistencias pull up de 10k Ohms para mantener una señal constante de un 1 lógico. El modelo de limit switch que utilizamos se puede observar en la figura 12.



Figura 12. Limit Switch.

### *E. Actuadores.*

Decidimos utilizar un motor de pasos bipolar modelo NEMA17, el cuál necesita ser alimentado por una fuente de voltaje de +12VDC, y consume 1.5A en cada bobina. Por cada paso del motor, se movía 1.8°, dando 200 pasos por rotación. Para controlarlo, se utilizó un driver A4988. El motor tiene 4 cables que conectan a sus 2 bobinas, en la figura 13 se observan sus 4 cables y respectivo orden de conexión (rojo con azul y negro con verde).

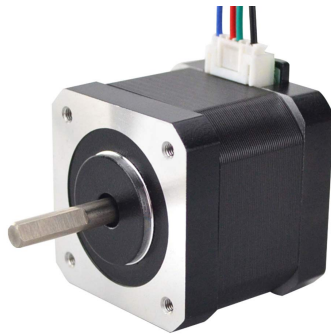


Figura 13. Motor de pasos NEMA17.

Para el control del motor de pasos se contó con un driver A4988, debido a su facilidad de uso y compatibilidad para manejar motores de pasos, la asignación de pines del driver A4988 se puede observar en la figura 5 las conexiones realizadas entre el arduino y el driver fueron las siguientes: el DIR pin (#11 del A4988) fue conectado al pin digital #8 del arduino el cual dependiendo de la señal que recibía, ya sea un alto o un bajo, cambiaba la dirección de movimiento del motor, el STEP pin (#10 del A4988) fue conectado al pin digital #4 del arduino el cual controlaba la cantidad de pasos que daba el motor al momento de girar, según la hoja de datos del motor, cada paso equivale a  $1.8^\circ$ , por lo que se requieren 200 pasos para dar una vuelta completa, el EN pin (#16 del A4988) fue conectado al pin #11 del arduino para desenergizar las bobinas del motor cuando no se necesite activarlas y por ende no sufran sobrecalentamientos ni se gaste energía, los pines RST (#12) y SLP (#11) se conectan entre sí para mantener activo al microcontrolador del driver.

Se conectó el pin #16 al positivo de la fuente de voltaje y tierra (pines #9 y #15) al negativo, mientras que el pin #10 sirvió como alimentación para el microcontrolador del driver, por lo que se alimentó con +5VDC proporcionados por el arduino. Observar figura 14.

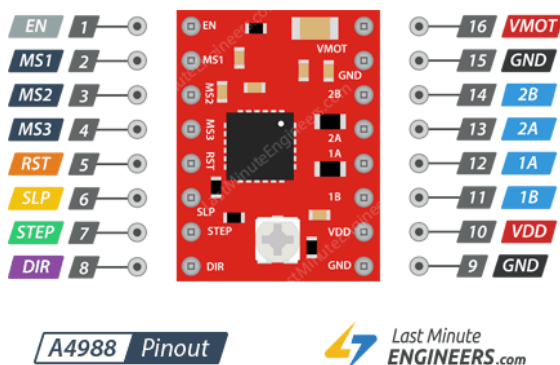


Figura 14. Asignación de pines del A4988.

Otro actuador incorporado en el circuito fue un solenoide JF-0306B, el cuál puede operar con 6V, 12V o 24V de corriente directa, contando con una fuerza máxima de 6N y un desplazamiento lineal de 10 mm, fue alimentado por +12VDC a 1A, el cuál se controló mediante un transistor TIP122 conectado al arduino.

## V. ELECTRÓNICA

### A. Diseño del circuito

Se optó por utilizar un Arduino Nano con un microcontrolador ATMEGA328P, debido a su facilidad de uso, tamaño compacto, y compatibilidad con demás componentes y módulos que utilizamos para el proyecto.

Las conexiones realizadas en el circuito se pueden observar en la figura 15, entre los componentes que utilizamos, se utilizó una resistencia de 1k Ohms de 1W como resistencia de la base del transistor, el transistor utilizado fue un par Darlington TIP122, el cuál al recibir una señal de alto por su base, permitía el flujo de corriente a través del solenoide, permitiendo que este se activase y empujara la pelota. La base del transistor se conectó al pin digital #12 del Arduino. También, se añadió un diodo 1N4007 entre el pin emisor del transistor y la parte positiva de la fuente de voltaje, de manera que la corriente inducida por el solenoide se regrese y no afecte ni al microcontrolador ni al circuito.

Asimismo, otra medida de protección que se utilizó en el circuito fue la incorporación de un capacitor en paralelo con la fuente de alimentación, se optó por un condensador electrolítico de 1000uF capaz de soportar hasta +25VDC, esto para crear un camino de baja impedancia para distorsiones de alta frecuencia, ruido o picos de voltaje que puedan entrar por la fuente de alimentación y afectar el funcionamiento del circuito.

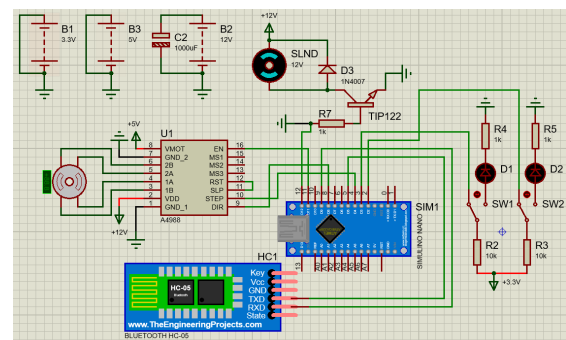


Figura 15. Esquemático de conexiones del circuito.



También se integraron un módulo bluetooth HC-05 para establecer comunicación entre el arduino y una aplicación creada para el control del robot, conectando los pines TX y RX del módulo a los pines digitales (PWM) #5 y #9 del arduino respectivamente, junto con un driver A4988 explicado más a detalle en el apartado de actuadores.

Se puede apreciar todo el circuito en protoboard en la figura 16:

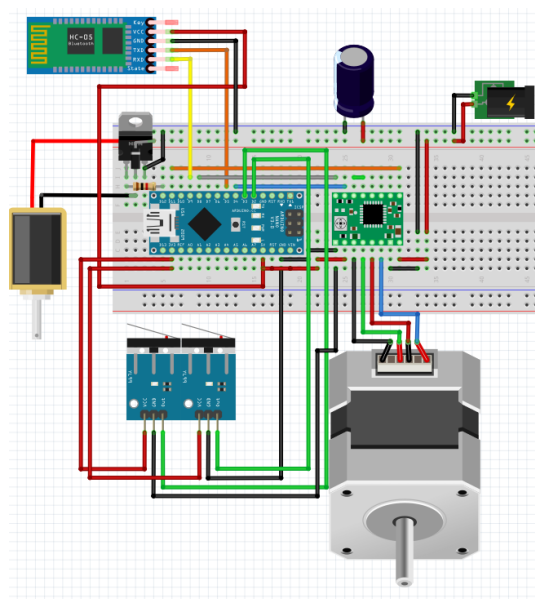


Figura 16. Conexiones del circuito en Fritzing.

En la figura 17, se observan las conexiones que realizamos en una protoboard de manera física.

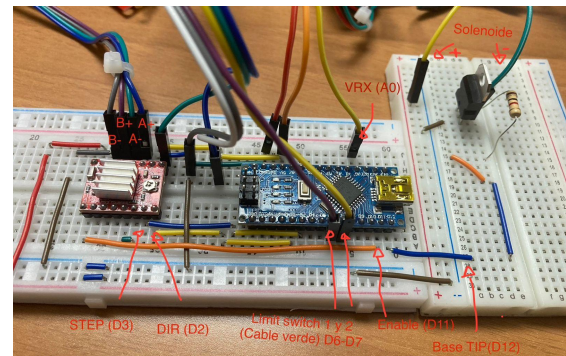


Figura 17. Conexiones en protoboard (Previo a la incorporación de módulo Bluetooth)

### B. Comunicación

Si el robot se comunica con otros dispositivos o sistemas, describir los componentes electrónicos utilizados para la comunicación, como módulos de comunicación inalámbrica (Wi-Fi, Bluetooth, Zigbee) o puertos de comunicación (USB, UART, SPI). Explicar cómo se establece la comunicación y qué protocolos se utilizan.

### C. Sistema de alimentación

El circuito fue alimentado por una fuente de voltaje que proporcionaba +12VDC a 1.5A, esto para alimentar al motor y al solenoide, mientras que para poder alimentar al arduino se utilizó un cable USB a mini USB (entrada del arduino Nano) enchufado a un adaptador de corriente que proporcionaba +5VDC a máximo 1 amperio de corriente. Estos permitieron que el circuito fuera alimentado directamente por el sistema eléctrico mediante enchufes, sin tener la preocupación de que se fuera a descargar el circuito.

## VI. ADMINISTRACIÓN DE PROYECTOS

### A. Metodología para la distribución de tareas

En este proyecto utilizamos la metodología SCRUM.

Scrum es una metodología de trabajo que se utiliza para abordar los desafíos comunes en la gestión de proyectos complejos, donde los ámbitos y circunstancias pueden cambiar de manera frecuente e impredecible.

Scrum se basa en principios de transparencia, inspección y adaptación. Su objetivo principal es facilitar la colaboración de los miembros del equipo, para lograr los objetivos establecidos. Observar *tabla 4*.



## B. Planeación de tiempos

Act.	Descripción	Time	Persona
Pedir comp.	Pedir en línea los componentes	30min	Leonardo
Diseños	Diseño en CAD de caja, robot, cuello, soportes etc.	6hrs	Todos
Impresión y cortes	Cortar MDF e imprimir piezas	10hrs	Miguel, Jonatan y Michell
Construcción de la electrónica	Conectar cables y componentes	2hrs	David
Construcción de la mecánica	Unir todos los tornillos, poleas, perfiles etc	1hr	Leonardo
Construcción de la caja	Unir las partes de la caja	20min	Erick
Creación de código	Redacción del código	3.5 hrs	Todos
Unión de todos los componentes	Unir mecánica, electrónica y programación	2hrs	Todos

Tabla 4. Actividades realizadas

## C. Resultados del sprint

### SPRINT 1

En el primer sprint concluimos con pedir los materiales y realizar el primer prototipo con todos los primeros diseños tanto de la caja como del robot. También en este primer sprint, comenzamos a investigar sobre los componentes que íbamos a utilizar para conocerlos mejor y probarlos. En la siguiente *figura 18* podemos observar las actividades específicas.

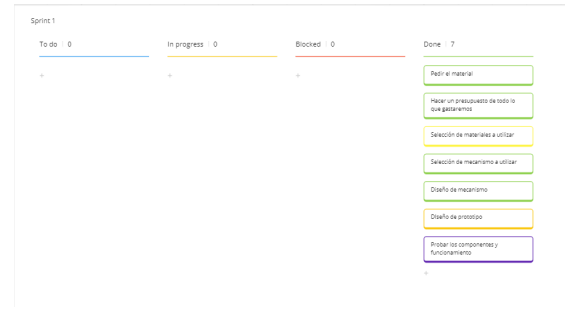


Figura 18. Sprint 1

### SPRINT 2

En el segundo sprint concluimos lo más “pesado” (*Figura 19*). Se trata de las actividades más importantes como la impresión de 3D y el corte del MDF.

Así también concluimos con todo el circuito eléctrico y todo el código de programación listos para ponerlos en práctica.

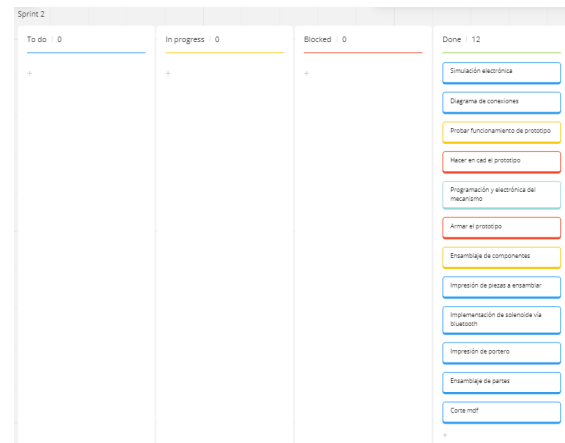


Figura 19. Sprint 2

### SPRINT 3

Finalmente en el sprint 3, tuvimos como resultado la presentación del robot final. Además, de la presentación de nuestra aplicación bluetooth. (*Figura 20*)

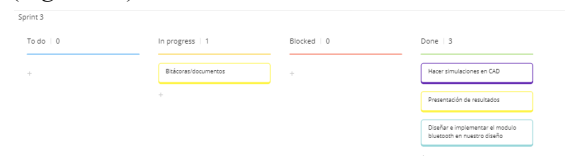


Figura 20. Sprint 3

## D. Ayuda de la planeación

La planeación nos resultó muy útil para la realización del proyecto pues no solo nos brindó la organización sino también nos ayudó a darle un seguimiento y un control a todas las actividades que debíamos realizar.

Con la ayuda de la planeación logramos tener una buena gestión del tiempo así como la priorización

de algunas actividades importantes y que de ellas dependían los demás.

Finalmente lo que nos brindó la organización correcta de todo es la asignación de tareas de manera equitativa, para que así todos estuviéramos pendientes y todos pudiéramos aprender un poco de todas las actividades.

Al tener todo ordenado, pudimos lograr nuestro objetivo final de la mejor manera, lo cual se ve reflejado en nuestra calificación y adquisición de conocimientos.

## VII. CONCLUSIONES

En Conclusiones individuales incluir ¿Cuál fue tu aportación individual al equipo? ¿Qué aprendiste que no tenías en el semestre pasado? ¿Cuál fue el mayor reto y cómo lo superaste?

### A. Erick Pérez

Dentro del reto me encargue de diseñar en CAD y mandar a cortar la caja donde irían situado todos los mecanismos, además hacía pequeñas aportaciones en lo que se necesitaba como pegar, cortar, armar, etc. Lo que pude aprender en este proyecto fue el diseñar una caja que iba a ser especialmente cortada en MDF ya que se tenía que hacer pieza por pieza, para después hacer un ensamble de ella. Creo que el mayor reto fue juntar toda la electrónica con la mecánica, porque no sabíamos muy bien como acomodar todo, esto lo solucionamos con una lluvia de ideas y una buena comunicación.

### B. Miguel Barragan

Durante la realización del proyecto, me encargué de diseñar en SolidWorks tanto la caja principal como el portero y el mecanismo de pegada. Utilizando esta poderosa herramienta de diseño asistido por computadora, pude dar vida a mis ideas y transformarlas en modelos tridimensionales detallados y precisos.

Comencé por diseñar la caja principal, que sería el contenedor central del proyecto. Teniendo en cuenta las especificaciones y requisitos. Consideré

aspectos como el tamaño, la forma, los ángulos y los materiales adecuados para garantizar la integridad y resistencia de la caja, al mismo tiempo que optimiza su eficiencia y estética.

Luego, me concentré en el diseño del portero, una parte clave del proyecto. Aprovechando las capacidades de modelado en 3D de SolidWorks, creé una representación virtual del portero con todos sus componentes y detalles. Asegurándose de considerar factores como el tamaño, la ergonomía y la funcionalidad, trabajé en estrecha colaboración con el equipo para iterar y mejorar continuamente el diseño hasta obtener un resultado satisfactorio.

Aprendí los fundamentos de la programación en Arduino y cómo utilizarlo para controlar dispositivos electrónicos. También adquirí habilidades en el montaje y conexión de circuitos mediante el uso de un protoboard, lo cual me permitió realizar pruebas y experimentos de manera efectiva. Además, me familiaricé con SolidWorks, una poderosa herramienta de diseño asistido por computadora, que me permitió crear modelos 3D precisos y detallados. También adquirí conocimientos en la administración de proyectos, aprendiendo a planificar, organizar y gestionar eficientemente los recursos y el tiempo. En general, mi aprendizaje abarca diferentes áreas, lo que me proporcionó una base sólida en electrónica y otras disciplinas relacionadas, ampliando así mi conjunto de habilidades y conocimientos.

Uno de los mayores desafíos que enfrenté fue la elaboración de la caja para el proyecto. Desde un principio, no revisamos detalladamente las restricciones y requerimientos, lo cual resultó en la necesidad de realizar varias iteraciones del diseño de la caja. Este proceso de prueba y error nos demandó tiempo y esfuerzo adicional, pero fue crucial para obtener un resultado final satisfactorio. Además, decidimos añadir un toque de diseño a la caja para lograr una presentación estética y atractiva.

Otra dificultad que encontramos fue la incorporación de una rampa dentro de la caja para que la pelota pudiera salir por un lateral. Esto requería una cuidadosa planificación y diseño, asegurándonos de que la rampa cumpliera su función sin afectar la estructura y estabilidad de la caja. Fue necesario hacer ajustes y pruebas para encontrar el ángulo y la forma adecuada de la rampa, asegurando que la pelota se desliza suavemente sin obstáculos.

### *C. David Torres*

Durante este proyecto me encargué de la realización y diseño de los circuitos a utilizar, mediante simulaciones en softwares como Proteus y Fritzing así como la integración de los componentes en una protoboard para formar el circuito electrónico del portero, y su respectiva fuente de alimentación para el funcionamiento óptimo del robot en todo momento. Aprendí sobre la incorporación de distintos drivers, así como su control, programación y aplicación, considero que el mayor reto fue la elección de los componentes y distintos errores con la programación que se presentaron a lo largo del desarrollo del proyecto, puesto que fueron muchas las variables que pudieron impactar el que el circuito funcionara o no funcionara cómo se deseaba en determinado momento, por lo que se tuvo que estar atentos a distintos detalles que pueden impactar al área electrónica en el robot.

### *D. Leonardo Galán*

Mi aportación en el equipo fue más que nada en la parte mecánica ya que tanto ayude con las ideas para la caja y el robot, ensamblé del prototipo final del robot ya con todo dentro, de aprendizajes me llevo sobretodo la parte del diseño en cad ya que era algo que no había visto tan a fondo, el cómo hacer un ensamble, hacer un engrane, las ecuaciones globales de solidworks etc, en la parte eléctrica me llevo el tema de la fuente de voltaje, como funciona y cómo es que se hace, para la parte de administración aprendí acerca de la metodología SCRUM hasta entonces desconocida para mí, me certifiqué en ello y lleve a cabo algunas actividades del equipo, por último pero no menos importante en la parte electrónica si bien ya tenía conocimientos no tan básicos de arduino hubo algunos que refresque y otros que aprendí de manera nueva y más eficiente a la que ya conocía. A nivel personal el mayor reto que me lleve fue el acomodarme mis tiempos al proyecto ya que hasta entonces no había tenido un proyecto tan demandante a nivel práctico y al inicio estaba un poco perdido con este aspecto pero con apoyo de mi equipo y de reflexionar en lo personal lo logré superar

### *E. Jonatan De la Rosa*

Mi aportación al equipo principalmente fue en el area de programacion , ya contaba con bastantes conocimientos en esta área por lo que me resulto facil y me sentí bastante agusto, por lo que me encargue de hacer la logica de programacion así como el código que se utilizó el día de la presentación y diferentes pruebas de sensores junto a David Torres como electrónico y a pesar de que tenía amplio conocimiento de programación puedo decir que aprendí sobre interrupts de los cuales tenia una nocion pero no los había aplicado como tal en un proyecto, de la misma forma también ayude en temas de impresiones 3d ya que me era fácil imprimir algunas piezas 3d por lo que imprimir también fue una de mis aportaciones, por último puedo decir que el reto más importante que me puso este proyecto fue el área del manejo del tiempo y planeación de proyectos fue algo que me llevo demasiado conmigo, el curso de SCRUM es una de las cosas que de la misma forma que me reto me enseñó la importancia de la planeación para una buena ejecución

### *F. Michell Sarmiento*

Mi aportación individual hacia el equipo fue el diseño de los soportes y de la caja. Así como de poner a imprimir la mayoría de los componentes que se necesitaban en 3D. Además, aporté a la realización de varios entregables de diversos módulos como reportes y tareas.

Con este reto me llevo bastantes aprendizajes, empezando por la programación. Anteriormente no conocía mucho sobre los temas, aprendí sobre variables, funciones, delays etc; además, junto con programación viene la electrónica de la cuál aprendí bastante especialmente del módulo bluetooth con el cuál no había trabajado. Normalmente, debido a mis preferencias, me gusta trabajar en la parte mecánica; sin embargo, con este reto pude abarcar mis conocimientos a estas 2 áreas que se relacionan las cuáles me gustaron bastante.

Mi mayor reto en este proyecto fue el autoaprendizaje. Por un lado, para este reto era necesario buscar más información de la proporcionada en clases para un mejor entendimiento y por ello fue un poco tedioso leer mucha información de diversas fuentes y tratar de entender temas más avanzados; sin embargo, pude superarlo al relacionar la información que ya conocía con la nueva y no adentrarme a más información de las cuáles no tengo las bases

suficientes para entenderlas aún ya que como consecuencia trae una confusión imparable. Otro de los retos, como mencioné anteriormente, fue salir de mi zona de confort (mecánica) para adentrarme a otras áreas casi desconocidas como programación y electrónica, pude superarlo debido a las clases que tomamos y también gracias al conocimiento de mis compañeros apasionados por estos dos temas.

## **VIII. LO QUE SE APRENDIÓ EN ELECTRÓNICA DE MANERA INDIVIDUAL**

### *A. Erick Carranza:*

Durante el modulo de electronica aprendí cómo funcionan los diodos y el tipo de materiales de los que están hechos, además aprendí un cómo funcionan las fuentes de voltaje, de igual manera pude aprender del funcionamiento básico de un multímetro, y el cómo es que se deben de medir las cosas. De igual manera, me llevo un poco de conocimiento sobre los semiconductores y sus propiedades que ayudan demasiado en la industria.

### *B. Miguel Barragán:*

A lo largo de la materia de electrónica, he aprendido sobre diversos temas fundamentales en el estudio de los dispositivos electrónicos. Comencé explorando los conductores, materiales que permiten el flujo libre de corriente eléctrica debido a su baja resistividad. Luego, me adentré en el mundo de los semiconductores, que poseen una conductividad intermedia entre los conductores y los aislantes. Aprendí que los aislantes son materiales con alta resistividad, que dificultan el paso de la corriente eléctrica. Además, profundicé en la resistividad, una propiedad de los materiales que determina su oposición al flujo de la corriente. También estudiamos los materiales tipo n y tipo p, que son semiconductores dopados con impurezas para crear regiones con exceso de electrones (n) o con ausencia de ellos (p).

Uno de los componentes más importantes que analicé fue el diodo ideal, un dispositivo de dos terminales que permite el flujo unidireccional de corriente eléctrica. Estudiamos el modelo ideal de diodo, que se basa en la aproximación de considerar al diodo como una unión de dos

materiales semiconductores diferentes. Exploramos las aplicaciones de los diodos en diversos campos, como la rectificación de corriente alterna, la protección contra polaridad inversa y la generación de señales de radiofrecuencia. Además, analizamos específicamente las aplicaciones de los diodos en los LEDs (Light-Emitting Diodes), dispositivos que emiten luz cuando se aplica una corriente eléctrica en la dirección adecuada. Estos LEDs tienen aplicaciones amplias, desde iluminación eficiente hasta pantallas electrónicas y señalización visual.

En resumen, el estudio de la electrónica me ha proporcionado una base sólida sobre conductores, semiconductores, aislantes, resistividad, materiales tipo n y tipo p, diodos ideales, modelo ideal de diodo, así como sus diversas aplicaciones, incluyendo los LEDs. Este conocimiento me ha permitido comprender mejor el funcionamiento de los dispositivos electrónicos

### *C. David Torres*

Durante este módulo pudimos aprender sobre componentes electrónicos y sus características, así como su integración para la creación de una fuente de voltaje, cómo los son los diodos de silicio que sirven para crear un puente rectificador, y cómo al unirse a un transformador de corriente alterna e integración de capacitores bypass en paralelo junto con un regulador de voltaje (LM317 por ejemplo), se puede crear una fuente de voltaje confiable para diferentes aplicaciones en proyectos electrónicos. También aprendimos sobre los fundamentos de los semiconductores, y cómo estas propiedades permiten la fabricación y aplicación de diodos, transistores y LEDs. Además, en este módulo puede incrementar mis habilidades de análisis de circuitos e identificación de componentes.

### *D. Leonardo Galán*

A lo largo de las 5 semanas en las cuales tuvimos este módulo aprendí algunos conceptos que si bien parecen básicos no tenía hasta el momento, como por ejemplo el tema de los diodos y como es que funcionan, en que se pueden aplicar y porque son importantes para los circuitos eléctricos, además el tema de poder hacer nuestra propia fuente de voltaje si bien no lo implementamos dentro de nuestro proyecto me lo llevo por si en algún momento lo necesito y sobretodo el aprendizaje de cómo funciona cada uno de los elementos de esta.

Por último, mirando en retrospectiva y como una crítica hacia mi persona me llevo el hecho de investigar acerca de lo básico del módulo ya que si bien salió el trabajo si batallé un poco en algunas cosas ya que no tenía el conocimiento previo necesario.

#### E. Jonatan De la Rosa

Este módulo además de los conocimientos adquiridos en estas 5 semanas, pude poner en práctica conocimientos teóricos de otras clases como lo son electromagnetismo, de la misma forma pude entrar mas en contacto con temas más apegados a la realidad como lo es un puente de diodos para tener una rectificación de corriente que aunque por cuestiones de tiempo no lo pudimos implementar el investigar y tratar de hacerlo te dan conocimientos que se pondrá en práctica en otras materias, también me agrado mucho la dedicación del profesor en querer la participación del grupo para mediciones de resistencias o de corrientes y es algo que me llevo de él.

#### F. Michell Sarmiento

En lo personal me gustó bastante el módulo de electrónica, esto debido a que algunos eran temas nuevos para mi y por tanto me causaban mucha curiosidad algunas cosas, incluso fuera de los temas.

Lo que me llevo de este módulo principalmente fue la introducción de los diodos los cuales son de gran utilidad para la rectificación de corriente (que fluya en una sola dirección). También son bastante útiles para estabilizar el voltaje, lo cual fue de gran ayuda para la creación del proyecto.

Finalmente con este módulo me llevo bastantes aprendizajes de cómo la electrónica vive con nosotros, pláticas que nunca había tenido en otras clases pues siempre eran problemas pero nunca lograba ver donde se aplicaban en la vida cotidiana hasta este módulo donde pude ver de manera realista los componentes electrónicos que no nos damos cuenta que utilizamos.

## IX. ANEXOS

**Código Github:** [Jonatandlr/Pruebas \(github.com\)](https://github.com/Jonatandlr/Pruebas)

### Código LimitS (Libreria de nosotros):

```
C/C++
#ifndef limitS_h
#define limitS_h
#include <Arduino.h>

class limitS{
    private:
        int _limitS;
        int _Value1;
    public:
        limitS(int limit){
            _limitS=limit;

        pinMode(_limitS,INPUT_PULLUP);
        };
        int value(){

            _Value1=digitalRead(_limitS);
            return(_Value1);

        };
};
#endif
```

### Código Principal:

```
C/C++
#include <limitS.h>
#include <SoftwareSerial.h>
#include <Arduino.h>

#define stepPin 4
#define dirPin 8
#define Enable 11
// #define limitL 2
// #define limitR 3

#define solenoide 12

int time=500;
int switchL_Value=1;
int switchR_Value=1;
```

```

int appValue;

//constructor del bluetooth
SoftwareSerial
miBT(5,9);//tx,rx del bluetooth
limitS limitL(2);
limitS limitR(3);

void limitInterruptL(){
  if (limitL.value()==0)
  {
    digitalWrite(Enable,HIGH);
    switchL_Value=0;
  }
  else{
    switchL_Value=1;
  }
}
void limitInterruptR(){
  if (limitR.value()==0)
  {
    digitalWrite(Enable,HIGH);
    switchR_Value=0;
  }
  else{
    switchR_Value=1;
  }
}

void setup (){
  //Pines motor
  pinMode(dirPin,OUTPUT);
  pinMode(stepPin,OUTPUT);
  pinMode(Enable,OUTPUT);

  //Pin Solenoide
  pinMode(solenoide,OUTPUT);

  attachInterrupt(digitalPinToInterrupt(2),limitInterruptL,CHANGE);

  attachInterrupt(digitalPinToInterrupt(3),limitInterruptR,CHANGE);

```

```

    errupt(3),limitInterruptR,CHANGE);
    Serial.begin(9600);
    miBT.begin(38400);
  }

  void loop(){

    if (miBT.available())
    {
      appValue=miBT.read();
      if (appValue==80)
      {
        digitalWrite(solenoide,HIGH);
      }
      else{
        digitalWrite(solenoide,LOW);
      }
      if
      ((appValue==76)&&(switchR_Value==1))
      {
        digitalWrite(Enable,LOW);

        digitalWrite(dirPin,HIGH);
        // Serial.println("a");
        while (appValue!=65){//83
          significa stop
          digitalWrite(stepPin,
            HIGH);

          delayMicroseconds(time);
          digitalWrite(stepPin,
            LOW);

          delayMicroseconds(time);
          appValue=miBT.read();
          if (appValue==80)
          {
            digitalWrite(solenoide,HIGH);
          }
        }
      }
    }
  }

```



```

        else if(appValue==83){

digitalWrite(solenoide,LOW);
        }
    }
    }
    else if
    ((appValue==82)&&(switchL_Value
==1))
    {
        digitalWrite(Enable,LOW);
        digitalWrite(dirPin,LOW);
        while
        (((appValue!=65))&&(switchL_Val
ue==1)){//83 significa stop
            digitalWrite(stepPin,
HIGH);

delayMicroseconds(time);
            digitalWrite(stepPin,
LOW);

delayMicroseconds(time);
            appValue=miBT.read();
            if (appValue==80)
            {

digitalWrite(solenoide,HIGH);
            }
            else if(appValue==83){

digitalWrite(solenoide,LOW);
            }
        }
    }
    else{

digitalWrite(Enable,HIGH);

```

```

        digitalWrite(stepPin,
LOW);
    }
}
else{
    digitalWrite(Enable,HIGH);

}
}

```

#### **Hojas de datos:**

##### **ATMEGA328P:**

<https://pdf1.alldatasheet.com/datasheet-pdf/view/241077/ATMEL/ATMEGA328P.html>

##### **Driver A4988:**

[https://www.pololu.com/file/0J450/a4988\\_DMOS\\_microstepping\\_driver\\_with\\_translator.pdf](https://www.pololu.com/file/0J450/a4988_DMOS_microstepping_driver_with_translator.pdf)

##### **Diodo 1N4007:**

<https://html.alldatasheet.com/html-pdf/207678/PANJIT/1N4007/215/1/1N4007.html>

##### **Módulo Bluetooth HC-05:**

[https://components101.com/sites/default/files/component\\_datasheet/HC-05%20Datasheet.pdf](https://components101.com/sites/default/files/component_datasheet/HC-05%20Datasheet.pdf)

##### **Motor NEMA17:**

<https://www.omc-stepperonline.com/download/17HS15-1504S-X1.pdf>

##### **Transistor Par Darlington NPN TIP122:**

<https://html.alldatasheet.com/html-pdf/652088/FAIRCHILD/TIP122/485/1/TIP122.html>

##### **Carpeta con anexos generales del equipo:**

<https://drive.google.com/drive/folders/1jBu30GIZPpbWpOssooQtG9ZYT4g2VEYB?usp=sharing>