

Universidad del Bío – Bío Facultad de Ciencias Empresariales
Ingeniería de Ejecución en Computación e Informática



UNIVERSIDAD DEL BÍO-BÍO

Proyecto de título: Prototipo de seguimiento en tiempo real implementado en Arduino y microcontrolador ESP8266

**Memoria para optar al título de ingeniero en ejecución
en computación e informática**

Autor: Diego Nicolás Merino Troncoso
Profesor Guía: Patricio Galdames Sepúlveda
Concepción, 1er Semestre 2018.

RESUMEN

El presente proyecto, consiste en el diseño, construcción y explicación de un sensor de seguimiento IMU (unidad de medida inercial), que permite mostrar los movimientos de un dispositivo con su aceleración y rotación hechos para dar apoyo a unas futuras mejoras de seguimiento y movimientos de personas (en sus extremidades o cuerpo).

El dispositivo construido permite observar los movimientos en tiempo real, sin conexión de cables, traspasando los datos de manera wifi a un computador. La investigación realizada en este proyecto da muy buenos resultados en pro de seguir con la mejora de dispositivos de mucho menor valor comercial para su fabricación, que otorga resultados alentadores para observar movimientos de personas en forma virtual.

La relevancia del estudio tiene relación con dar respuesta a las necesidades y mejoras tecnológicas que se pueden llevar a cabo de una manera más económica y, de esta manera, poder llegar a más personas, con una mejor atención en centro de atención motora, y por consiguiente, llegar a pensar en dispositivos portátiles que guarden información de los movimientos de las personas en una tarjeta SD, la cual pueda ser reproducida en una aplicación por un especialista y ver los movimientos realizados por el paciente.

Finalmente, con los resultados obtenidos y aquí expuestos, el proyecto incita a seguir con la creación de dispositivos que muestren movimientos en tiempo real, cumpliendo con los requerimientos iniciales del área (movimiento en tiempo real, conexión wifi, dispositivo móvil [IMU]).

Índice

RESUMEN	2
INTRODUCCION	5
1 DESCRIPCION DE LA EMPRESA	6
1.1 Descripción del área de estudio	7
1.2 Descripción de la problemática	7
2 DEFINICION DEL PROYECTO	7
2.1 Análisis del problema	8
2.2 Objetivos generales del área	9
2.3 Objetivos del proyecto	10
2.4 Ambiente de ingeniería de software	10
3 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE.....	15
3.1 Alcances	15
3.2 Objetivos del software	16
3.3 Descripción global del proyecto	16
4 Estudio de factibilidad	32
4.1 Factibilidad técnica	32
4.2 Factibilidad Económica	33
4.3 Factibilidad operativa	36
4.4 Conclusión de la factibilidad	36
5 CONCLUSIONES	37
6 BIBLIOGRAFIA DIGITAL	38

Índice de figuras

Figura 1: Internet de las cosas	14
Figura 2: Esquema general del proyecto	15
Figura 3: Esquema sensor	17
Figura 4: Arduino	19
Figura 5: Conexión I2C.....	22
Figura 6: ESP8266.....	23
Figura 7: Esquema ESP8266 a computador	24
Figura 8: Processing.....	25
Figura 9: Datos del sensor.....	26
Figura 10: Interpretación de datos por processing.....	27
Figura 11: Arduino y MPU-6050.....	27
Figura 12: Arduino, MPU-6050 y ESP8266.....	28
Figura 13: MPU-6050 y ESP8266 conectados.....	29
Figura 14: Conexión a batería externa.....	29
Figura 15: Introducción de dispositivos en caja.....	30
Figura 16: Prototipo 0.1 completo con caja y quincha.....	31

INTRODUCCION

En la actualidad, las herramientas tecnológicas para medir comportamientos corporales de personas pre adulto mayor se utilizan en un porcentaje muy bajo de la población, ya que se usa mayormente en el área de la kinesiología. Sin embargo, estas herramientas son muy costosas y conllevan mucho tiempo por pacientes (aproximadamente 45 minutos), es por ello que no se puede llegar a atender la mayor cantidad de personas que idealmente se quisieran atender.

El enfoque del presente proyecto es crear e incentivar la creación de dispositivos de bajo costo, capaces de leer y mostrar el movimiento corporal, lo cual se vea reflejado en una aplicación, para complementar la evaluación de los especialistas y así, grabar sus movimientos en una posible mejora de este proyecto.

Este informe tiene por objetivo explicar y demostrar la creación de un dispositivo Unidad de Medida Inercial (IMU) mediante aplicaciones de placas Arduino uno, placa MPU 6050, ESP8266, las cuales pueden interpretar datos como aceleración y giroscopio, además de enviar datos mediante wifi y comunicarse con una aplicación (Processing) instalada en un computador, para mostrarlos como movimientos de una articulación corporal (brazo).

Es importante aclarar que este sensor de seguimiento no alcanza el rendimiento ni la calidad de seguimiento como productos profesionales, no obstante, aún es útil para diversas tareas, en las que la latencia o la derivada no son una gran problemática y sí puede mostrar movimiento en tiempo real.

1 DESCRIPCION DE LA EMPRESA

El proyecto que a continuación se expone en el presente informe, se desarrolló para la empresa SmartGrows, cuyo rubro corresponde a el desarrollo tecnológico de proyectos de ingeniería. La empresa se ubica en Venecia 300, chiguayante.

A continuación, en este capítulo se realizará una descripción de la empresa para conocer más aspectos de ella y dar a conocer algunos detalles del área a trabajar y problemática del porque hacer el proyecto.

Empresa

Nombre: SmartGrows
Dirección: Venecia 300, Chiguayante
Rubro: Proyectos de ingeniería
Teléfono: +56992915917

Visión

Transformarse en una empresa de renombre tecnológico de la región, con un equipo multidisciplinario de jóvenes emprendedores que cuentan con la formación profesional, las ganas y el conocimiento adecuado para ofrecer un servicio completo a sus clientes.

Misión

Llevar la última tecnología disponible a nuestros clientes, ofreciendo una solución innovadora y práctica para satisfacer sus necesidades y diferenciarlo de sus competidores.

1.1 Descripción del área de estudio

El área de estudio corresponde al sector Arduino y ESP, enfocado al sensor IMU que es una unidad de medición inercial (del inglés inertial measurement unit), dispositivo electrónico que mide e informa acerca de la velocidad, orientación y fuerzas gravitacionales de un aparato, usando una combinación de acelerómetros y giróscopos.

1.2 Descripción de la problemática

En la actualidad, el sistema de implementación de un sistema IMU es tremendamente costoso, los artefactos que se encuentran a la venta son específicos para áreas determinadas, como video juegos o usos de tratamientos médicos, sin embargo, muchas veces superan el presupuesto de un ciudadano promedio.

Por lo anterior, uno de los pilares fundamentales de este proyecto es el bajo costo que éste trae consigo, y dejar abierta la investigación para hacer proyectos a gran escala a precios muy reducidos.

2 DEFINICION DEL PROYECTO

En este capítulo se presentan todos los antecedentes estudiados e investigados para llevar a cabo este proyecto, dando a conocer los requerimientos análisis y especificaciones de cada área para llevar a cabo este proyecto. Y en base a esto, desarrollar la propuesta de prototipo.

2.1 Análisis del problema

A continuación, se realiza un análisis del problema para ayudar en su entendimiento y comprensión.

Situación actual

El área donde se desarrolla este proyecto será de programación Arduino, ésta contempla sensores, conexión TCP/IP y programación en processing, específicamente en el área de prototipado de la empresa. Para este proyecto solo trabaja una persona, la cual desarrolla todas las labores mencionadas.

El problema radica en el costo y la poca comodidad de los sensores de seguimiento o movimiento disponibles en el mercado, la mayoría de estos sensores están enfocados al área de videojuegos, a partes específicas de la bioingeniería y otros, estos son muy lejanos al común de las personas y un tratamiento de seguimiento kinesiológico consta de 24 sensores cada uno, avaluado en cerca de \$500.000 mil pesos, es por ello que solo se encuentra en clínicas, hospitales o universidades. Estos en sí, son más específicos con respecto a entrega de resultados y además reflejan los movimientos en tiempo real con necesidad de una cámara y seguimiento específico de cada sensor, por lo mismo son muy incómodos, lejanos al común de las personas y costosos en caso de intentar mejorar el sistema existente.

El estudio y las investigaciones que se han realizado para crear dispositivos paralelos a ellos son muy favorables, teniendo en cuenta el bajo costo que implica hacer dispositivos Arduino o con otros tipos de microcontroladores y sensores.

Actualmente, con la tecnología médica existente, el tiempo de atención a un paciente es de 45 minutos aproximadamente, en lo que es instalar sensores, ubicar

al paciente, que éste camine y todo lo que esto conlleva, por tanto, el proceso resulta tedioso e incómodo para personas que presentan dificultades motoras.

Debilidades

En la actualidad, los sistemas son muy eficaces y específicos, además de estar comprobados por la comunidad científica. Reemplazarlos no es la idea futura del proyecto, sino que complementar para casos que no sean tan específicos y que puedan ayudar al seguimiento motor de las personas.

Probablemente nunca llegue a reemplazar ni sea tan exacto como el sistema actual, ya que tiene un costo bajo, además existen equipos que llevan varios años mejorando esto a nivel mundial, pero si da la posibilidad de un complemento para ayudar al profesional en su tiempo de gestión y costos.

El prototipo creado no corresponde a un prototipo final con las conexiones a todo un torso inferior o superior, sino tan solo un brazo, pero con la idea a mejorar de lo factible de un seguimiento mediante prototipado Arduino.

2.2 Objetivos generales del área

- Mostrar movimientos en tiempo real del prototipo
- Hacer un prototipo móvil y de fácil uso
- Cumplir con los plazos de iteraciones establecidos con la empresa
- Llevar un correcto seguimiento en informe de los avances y alcances del proyecto

Principales funciones del área

- Captura de datos
- Codificación de datos
- Interpretación de datos
- Prototipo

2.3 Objetivos del proyecto

Objetivo General:

Desarrollar un prototipo que opere con microcontrolador ESP8266 y ATmega328p (Arduino uno), que permita mostrar los movimientos en tiempo real de una determinada extremidad u objeto.

Objetivos específicos:

- Investigar sobre los lenguajes de programación que utilizan los dispositivos.
- Estudiar el uso de Arduino en el campo de la lectura de movimientos.
- Investigar sobre el uso de implementación del tipo de sensor a utilizar para la captura de datos.
- Hacer que el dispositivo sea lo más simple posible y fácil de usar para las personas.
- Hacer que el proyecto se lo mas económico y eficiente posible.
- Crear un modelo a escala para demostración final del proyecto

2.4 Ambiente de ingeniería de software

2.4.1 Metodologías de desarrollo

El proyecto es desarrollado utilizando el método de desarrollo iterativo-incremental, las iteraciones serán 1, debido al tiempo de entrega. Las iteraciones se pueden entender como mini proyectos: en todas las iteraciones se repite un proceso de trabajo similar para proporcionar un resultado completo sobre el producto final.

Metodología iterativa incremental: Esta metodología nos permite realizar una evolución sostenida de la solución, prácticamente desde el primer momento, teniendo en cuenta la solución seleccionada. Durante las distintas iteraciones se suministra al cliente funcionalidad adicional, construyendo así un entorno adecuado para el cumplimiento de los requerimientos, en ocasiones cambiantes.

Beneficios:

- Se pueden gestionar las expectativas del cliente de manera regular, puede tomar decisiones en cada iteración. Esto es interesante cuando: el cliente no sabe exactamente lo que necesita, cambios a coroto plazo (requisitos o cambios ya realizados).
- El cliente puede obtener resultados importantes y usables ya desde las primeras iteraciones.
- Permite conocer el progreso real del proyecto
- Permite mitigar desde el inicio los riesgos del proyecto

- El cliente como máximo puede perder los recursos dedicados a una iteración, no los de todo el proyecto

Restricciones

- Depende de la disponibilidad del cliente, ya que el cliente debe prácticamente participar en cada iteración del proyecto
- Es necesario disponer de técnicas y herramientas que permitan hacer cambios fácilmente en el producto, de manera que pueda crecer en cada iteración de manera incremental sin hacer un gran esfuerzo adicional, manteniendo su complejidad minimizada y su calidad
- Cada iteración ha de aportar un valor al cliente, entregar unos resultados cerrados que sean susceptible de ser utilizados por él

2.4.2 Técnicas y notaciones

Para el desarrollo del proyecto se utilizará lo siguiente:

- Para la toma de requerimientos se realizan entrevistas con el cliente que necesita del prototipo.

2.4.3 Estándares de documentación, producto y proceso

- Estándar para el informe: Plantilla “Planilla de documentación del proyecto de Desarrollo de Software” versión 15 de noviembre de 2011.
- IEEE Software Test Documentation Std 829 -1998

2.4.4 Herramientas de apoyo

- Editor de texto Sublime Text 3.
- Processing
- Aplicación Arduino IDE
- Libre Office Writer

2.4.5 Definiciones, siglas y abreviaciones

IMU: Unidad de medida inercial.

Microcontrolador: es un circuito integrado programable, capaz de ejecutar ordenes grabadas en su memoria.

Arduino: Es una plataforma de prototipos electrónica de código abierto, basada en hardware y software.

Arduino r3: Arduino que utiliza microcontrolador ATmega328.

MPU 6050: Es una unidad de medida inercial de 6 grados de libertad, pues combina un acelerómetro de 3 ejes y un giroscopio de 3 ejes.

ESP8266: Es un puente de puerto serie a WIFI, incluye un microcontrolador para manejar el protocolo TCP/IP.

Sensor: Dispositivo que capta magnitudes físicas (variaciones de luz, temperatura, sonido, movimiento etc.) u otras alteraciones del entorno.

2.4.6 Internet de las cosas (IoT)

La internet de las cosas es un concepto que se refiere a la interconexión digital de objetos cotidianos con internet.

Desde hace unos 30 años que se viene trabajando con la idea de hacer un poco más interactivos todos los objetos de uso cotidiano. Ideas como el hogar inteligente, también conocido como la casa del mañana, han evolucionado antes de que nos demos cuenta en el hogar conectado para entrar al Internet de las cosas.

El Internet de las cosas potencia objetos que antiguamente se conectaban mediante circuito cerrado, como comunicadores, cámaras, sensores, y demás, y les permite comunicarse globalmente mediante el uso de la red de redes.

A diferencia de algunas tecnologías mucho más populares entre las masas, el Internet de las cosas no ha encontrado su foco de explosión en el mercado del consumo. Quizás la tecnología está aún demasiado verde, o quizás los grandes del sector no han visto la oportunidad correcta para abalanzarse encima. Aun así, hemos visto como Apple y Google han dado algunos pasos discretos con tecnologías como Home Kit y Android @Home.

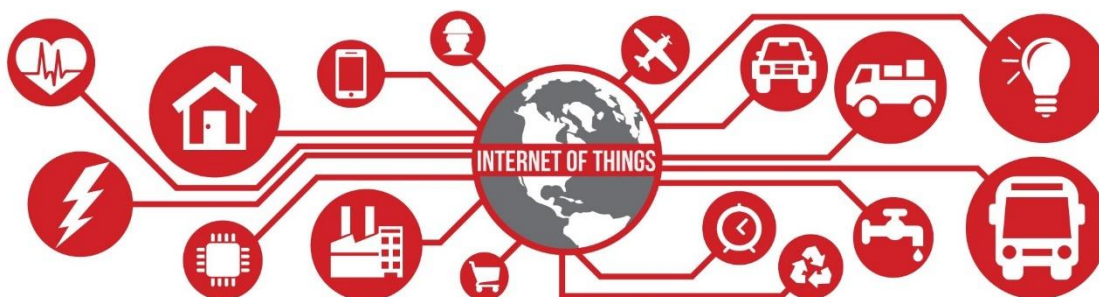


Figura 1. Internet de las cosas

3 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE Y HARDWARE

Este capítulo, nos permitirá entender y establecer los requerimientos de hardware y software, junto con los alcances del proyecto, esto nos ayudará para plantear los objetivos con claridad, para lograr desarrollar un prototipo que cumpla con nuestros objetivos especificados en el proyecto.

A continuación, se detallarán los requerimientos de sistema y de hardware, donde se detallará cada elemento a utilizar y la tipología de red.

3.1 Alcances

Se creará un prototipo, que pueda mostrar los movimientos en tiempo real sin ningún tipo de conexión cableada a un computador, los usuarios podrán ver estos movimientos en una aplicación.

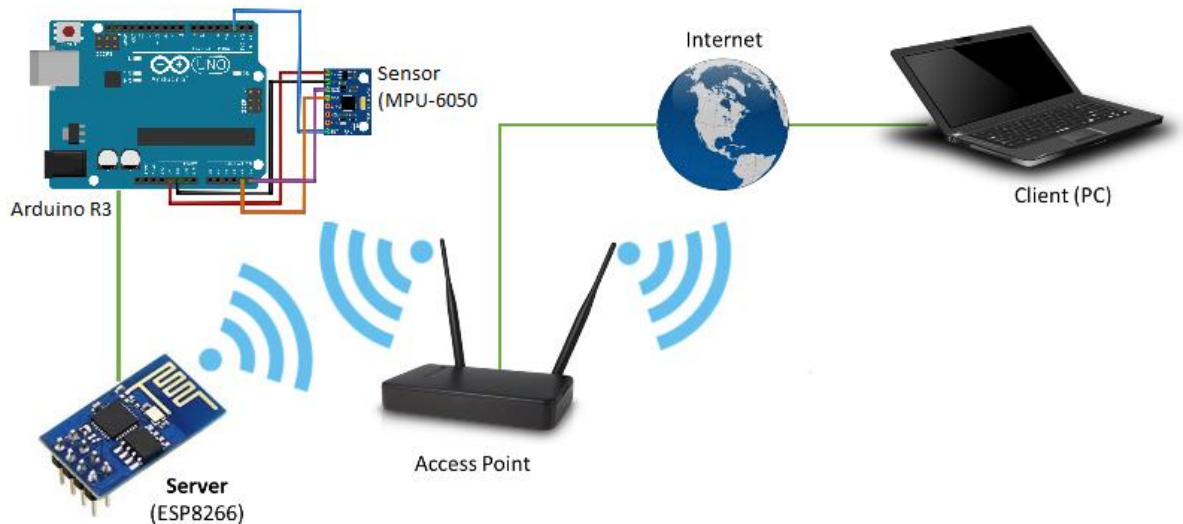


Figura 2. Esquema del proyecto general

3.2 Objetivos del software

El software utilizado se llama processing, el cual debe ayudar principalmente a gestionar los datos recibidos desde el sensor, Arduino y microcontrolador ESP, recibir datos mediante una conexión TCP y traducirlos en movimientos de una imagen previamente programada.

Objetivos específicos.

- Transformar la entrega de datos de forma serial a transmisión TCP/IP.
- Generar un archivo de texto con los datos leídos desde el sensor.
- Crear imagen que represente los movimientos del sensor.
- Mostrar los movimientos en tiempo real del prototipo.

3.3 Descripción global del proyecto

3.3.1 Introducción a sensores

Un sensor es todo aquello que tiene una propiedad sensible a una magnitud del medio, y al variar esta magnitud también varía con cierta intensidad la propiedad, ósea que nos manifiesta la presencia de dicha magnitud, y también su medida.

Un sensor consta de dos partes principales:

- un elemento sensible o activo, cuyas propiedades físicas o químicas cambian en presencia de aquello que se desea detectar.
- un elemento transductor, que convierte los cambios en las propiedades del elemento activo en una señal eléctrica.

Los sensores son una parte muy importante para la instrumentación y el control de los procesos industriales, se utilizan para poder determinar el estado del proceso donde están instalados, Ellos transforman las variaciones de la magnitud a medir en una señal eléctrica acondicionada de tal manera que pueda ser recibida en su destino.

La señal de salida de un sensor por lo general va a un indicador, a un registrador o a un controlador. Existen una gran variedad de sensores según la variable que se quiera medir, por ejemplo: presión, temperatura, nivel, flujo, posición (proximidad), velocidad, peso, voltaje, corriente, frecuencia, viscosidad, resistividad, radiación, pH, conductividad eléctrica, humedad entre otras.

Cuando los sensores tienen salidas digitales (sólo dos posibles valores) son llamados interruptores y cuando tienen salidas analógicas (más de dos posibles valores) son llamados transmisores.

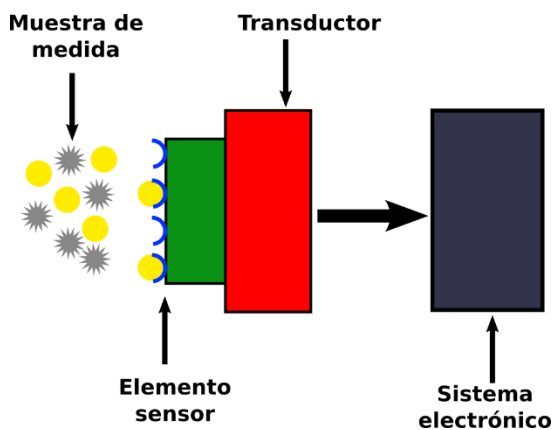


Figura 3. Esquema sensor

Características de los sensores

Muchas de las características de los sensores dependen de la variable a medir, pero otras son comunes a todos los sensores. Algunos de los aspectos a tener en cuenta en el momento de seleccionar un sensor son los siguientes:

- Exactitud: especifica la diferencia entre el valor medido y el valor real de la variable que se está midiendo.
- Resolución: es el cambio más pequeño que se puede medir.
- Precisión se compone de las características de conformidad y resolución.
- Sensibilidad: viene dado por el mínimo valor de la variable medida que produce un cambio en la salida.
- Error o latencia: es la desviación entre valor verdadero y valor medido.
- Rango es la diferencia entre el mayor valor y el menor valor que se puede medir.
- Rapidez de respuesta: capacidad del instrumento de seguir las variaciones de la entrada

3.3.2 Arduino



Figura 4. Arduino

Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar. Está pensado para artistas, diseñadores, como hobby y para cualquiera interesado en crear objetos o entornos interactivos.

Arduino puede sentir el entorno mediante la recepción de entradas desde una variedad de sensores y puede afectar a su alrededor mediante el control de luces, motores y otros artefactos. El microcontrolador de la placa se programa usando el *Arduino Programming Language* (basado en Wiring) y el *Arduino Development Environment* (basado en Processing). Los proyectos de Arduino pueden ser autónomos o se pueden comunicar con software en ejecución en un ordenador (por ejemplo, con *Flash*, *Processing*, *MaxMSP*, etc.).

Las placas se pueden ensamblar a mano o encargarlas pre ensambladas; el software se puede descargar gratuitamente. Los diseños de referencia del hardware (archivos CAD) están disponibles bajo licencia open-source, por lo que eres libre de adaptarlas a tus necesidades.

Características de Arduino

- **Barato:** Las placas Arduino son relativamente baratas comparadas con otras plataformas microcontroladoras. La versión menos cara del módulo Arduino puede ser ensamblada a mano, e incluso los módulos de Arduino pre ensamblados cuestan menos de 5500 pesos.
- **Multiplataforma:** El software de Arduino se ejecuta en sistemas operativos Windows, Macintosh OSX y GNU/Linux. La mayoría de los sistemas microcontroladores están limitados a Windows.
- **Entorno de programación simple y claro:** El entorno de programación de Arduino es fácil de usar para principiantes, pero suficientemente flexible para que usuarios avanzados puedan aprovecharlo también.
- **Código abierto y software extensible:** El software Arduino está publicado como herramientas de código abierto, disponible para extensión por programadores experimentados. El lenguaje puede ser expandido mediante librerías C++, y la gente que quiera entender los detalles técnicos pueden hacer el salto desde Arduino a la programación en lenguaje AVR C en el cual está basado. De forma similar, puedes añadir código AVR-C directamente en tus programas Arduino si quieres.
- **Código abierto y hardware extensible:** El Arduino está basado en microcontroladores ATMEGA8 y ATMEGA168 de Atmel. Los planos para los módulos están publicados bajo licencia Creative Commons. Incluso usuarios relativamente inexpertos pueden construir la versión de la placa del módulo para entender cómo funciona y ahorrar dinero.

3.3.3 Arduino uno r3

Este es el nuevo Arduino Uno R3 utiliza el microcontrolador ATmega328. En adición a todas las características de las tarjetas anteriores, Esto permite ratios de transferencia más rápidos y más memoria. No se necesitan drivers para Linux o Mac (el archivo inf para Windows es necesario y está incluido en el IDE de Arduino).

La tarjeta Arduino Uno R3 incluso añade pins SDA y SCL. Es más, hay dos nuevos pines cerca del pin RESET. Uno es el IOREF, que permite a los shields adaptarse al voltaje brindado por la tarjeta. El otro pin no se encuentra conectado y está reservado para propósitos futuros. La tarjeta trabaja con todos los shields existentes y podrá adaptarse con los nuevos shields utilizando esos pines adicionales.

Características

- Microcontrolador ATmega328.
- Voltaje de entrada 7-12V.
- 14 pines digitales de I/O (6 salidas PWM).
- 6 entradas análogas.
- 32k de memoria Flash.
- Reloj de 16MHz de velocidad.

3.3.4 Explicación de protocolo I2C

I2C es un bus de comunicaciones en serie.

I2C es un protocolo síncrono. I2C usa solo 2 cables, uno para el reloj (SCL) y otro para el dato (SDA). Esto significa que el maestro y el esclavo envían datos por el mismo cable, el cual es controlado por el maestro, que crea la señal de reloj. I2C no utiliza selección de esclavo, sino direccionamiento.

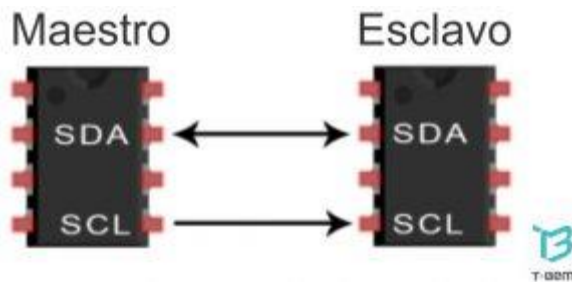


Figura 5. Conexión I2C

La principal característica de I2C es que utiliza dos líneas para transmitir la información: una para los datos y otra para la señal de reloj. También es necesaria una tercera línea, pero esta sólo es la referencia (masa). Como suelen comunicarse circuitos en una misma placa que comparten una misma masa esta tercera línea no suele ser necesaria.

Las líneas se llaman:

- SDA: datos
- SCL: reloj
- GND: tierra

Síncrona: esto quiere decir que el envío de bits por la vía de comunicación SDA está sincronizado por una señal de reloj que comparten tanto el maestro como el esclavo a través de la vía SCL

3.3.5 Microcontrolador ESP8266(modulo wifi)

Se trata de un chip integrado con conexión Wifi y compatible con el protocolo TCP/IP. El objetivo principal es dar acceso a cualquier microcontrolador a una red.

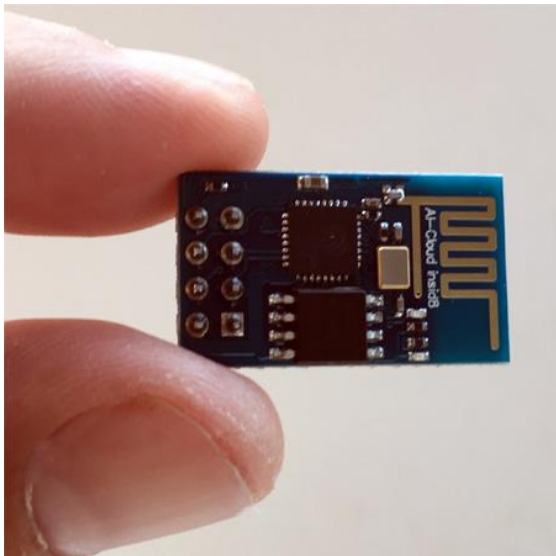


Figura 6. ESP8266

Hardware

- Utiliza una CPU Tensilica L106 32-bit
- Voltaje de operación entre 3V y 3,6V
- Corriente de operación 80 mA
- Temperatura de operación -40°C y 125°C

Conectividad

- Soporta IPv4 y los protocolos TCP/UDP/HTTP/FTP
- No soporta HTTPS en un principio. Sí que lo hace mediante software tanto en cliente como servidor TLS1.2. La primera implementación está todavía en desarrollo.

3.3.6 Envió de datos

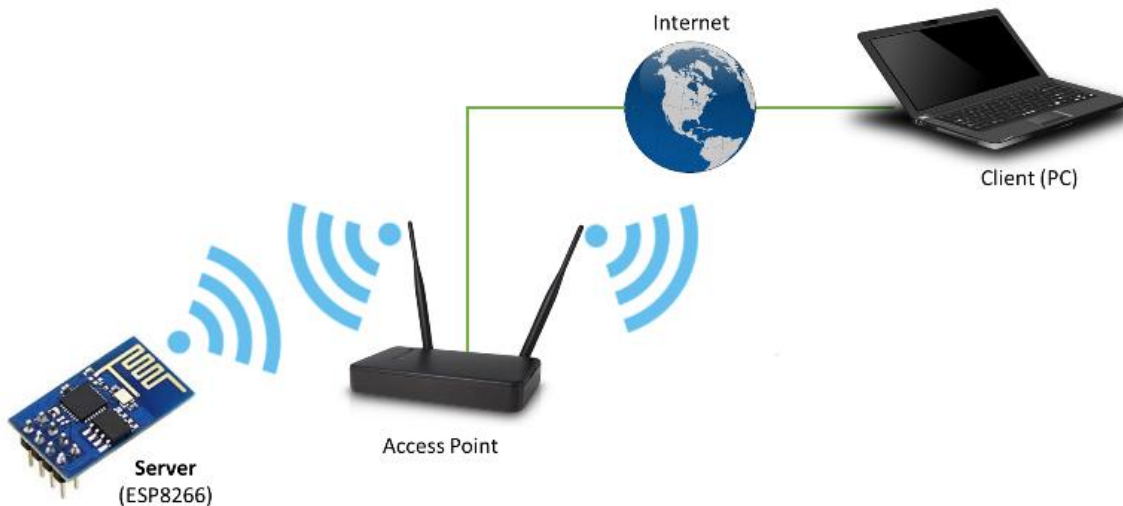


Figura 7. Esquema ESP8266 a computador

3.3.7 Socket TCP dentro de la misma red

TCP: Protocolo de control de transmisión, es uno de los protocolos fundamentales en internet. TCP es un protocolo orientado a conexión, es decir, que permite que dos máquinas que están comunicadas controlen el estado de la transmisión. En el nivel de aplicación, posibilita la administración de datos que vienen del nivel más bajo del modelo, o van hacia él, (es decir, el protocolo IP).

Arduino lee lo que viene por I2C y lo muestra por serial, pero lo que nosotros queremos hacer es enviarlo hacia un computador para esto ocupamos un microcontrolador ESP8266, ¿Qué es lo que? Se conecta a una red WIFI(router), conectamos a los pines RX, y TX para que lea los datos de arduino y no lo muestre por pantalla si no que llegue al microcontrolador(ESP8266), y este lo envíe mediante WIFI a un computador.

¿Cómo lo hace?

En realidad, ESP8266 posee una IP, y abre un socket TCP. Uno se puede conectar al él, en el puerto 23, ya que crea un servidor que te deja el puerto 23 abierto.

¿Qué pasa cuando se conecta a ese puerto 23?

Re direcciona todo lo que viene del serial al puerto, y todo lo que entra por el puerto al serial, ósea también se le podrían mandar comandos al Arduino por TCP.

3.3.8 Aplicación processing, el lenguaje para gráficos.



Figura 8. Processing

Processing es un dialecto de Java que fue diseñado para el desarrollo del arte gráfico, para las animaciones y aplicaciones gráficas de todo tipo. Desarrollado por artistas y para artistas.

Es un software basado en Java y, por lo tanto, multiplataforma.

Es una plataforma que integra entorno de desarrollo y lenguaje de programación.

La aplicación processing lee normalmente desde el serial de Arduino, pero lo modifique mediante código para que recibiera los datos por un socket

TCP, entonces, lee los mismos datos del serial, pero mediante una conexión TCP.

3.3.9 Requerimientos/ Restricciones de interfaz de comunicación

Se utilizará el protocolo I2C para la comunicación del arduino con los sensores, y el protocolo TCP/IP para el transporte de los paquetes de datos correspondientes.

3.3.10 Interpretación de datos

ypr	-1.08	29.23	-13.42
\$\$\$\$Á\$B:1bB			
ypr	-1.22	29.28	-13.36
\$\$\$\$ÁvB?1bB			
ypr	-0.34	29.53	-12.82
\$\$\$\$Á?B:1TB			
ypr	0.28	29.80	-12.68
\$\$\$\$ÁvB:1TB			
ypr	1.01	30.27	-12.95
\$\$\$\$Á°Bv10B			
ypr	1.77	30.73	-13.80
\$\$\$\$;ÁYBf1²B			
ypr	2.03	30.44	-15.46
\$\$\$\$?Á+B=1TB			
ypr	2.38	29.83	-17.11
\$\$ Á\$\$\$\$10B			
ypr	2.79	29.41	-18.19
\$\$			
\$\$\$\$^10B			
ypr	3.19	29.52	-18.52
\$\$			
XÁ B=16B			
ypr	3.63	29.79	-18.58
\$\$			
uÁ++10B			
ypr	4.54	30.01	-18.77
\$\$			
~Á-10B			
ypr	4.60	29.80	-19.99
\$\$\$\$PÁT010B			
ypr	4.16	29.13	-21.71
\$\$+~ÁqB<1TB			
ypr	3.97	28.64	-23.59
\$\$\$\$Á?B?1TB			
var	3.63	28.00	-25.66

Figura 9. Datos del sensor

Los Datos de la imagen anterior, son los datos que entrega el Sensor a Arduino que son retransmitidos a la aplicación processing mediante el microcontrolador ESP8266, a través de una señal WIFI, gracias al protocolo TCP/IP.

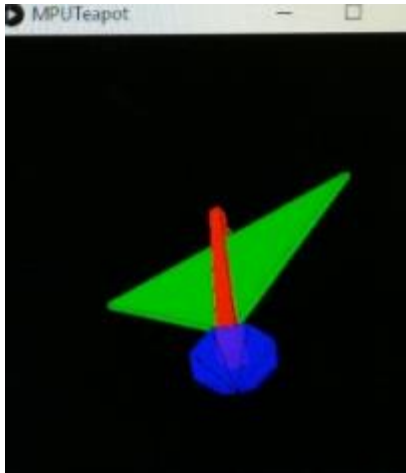


Figura 10. Interpretación de datos por processing

Los datos del sensor son interpretados como coordenadas de movimiento y posición por processing, la cual dibuja mediante código, unas formas previamente programadas de prueba.

3.3.11 Pasos para la construcción del prototipo

En esta etapa explicare paso a paso la construcción del prototipo 0.1

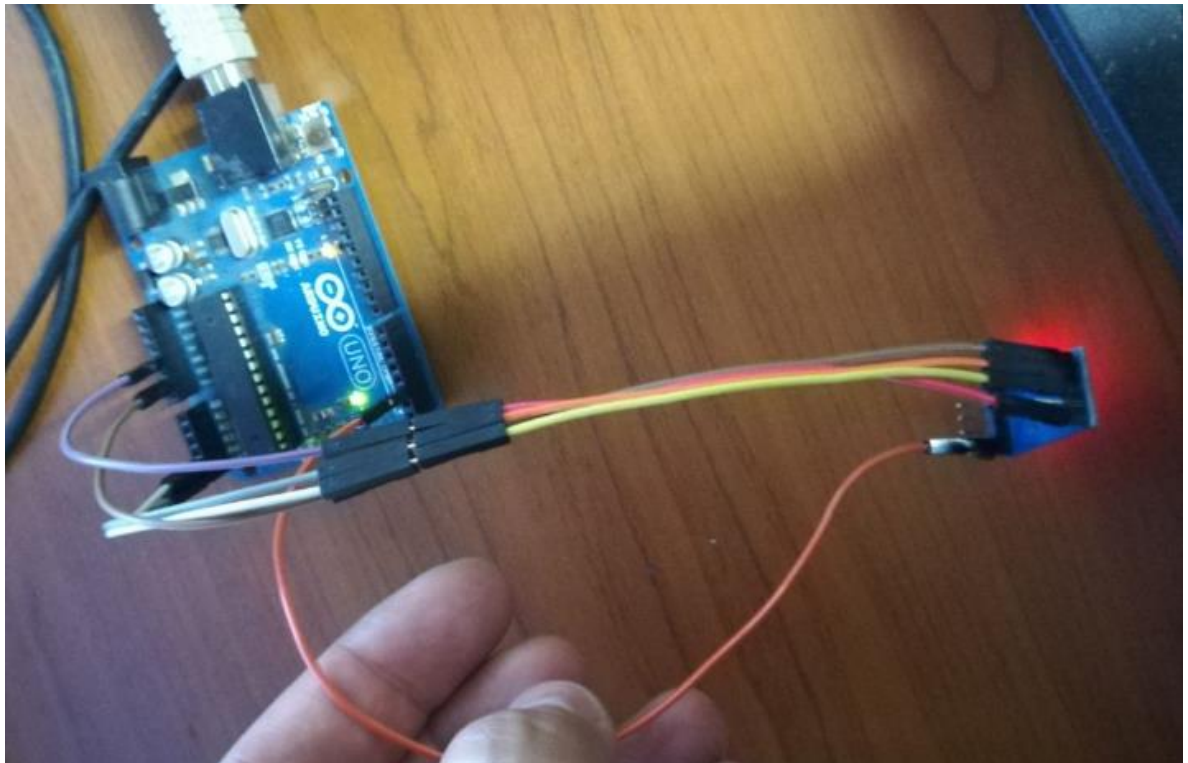


Figura 11. Arduino y MPU-6050

Lo primero es tener los implementos previamente mencionados para realizar el proyecto.

Comenzamos con la conexión y programación en Arduino del sensor MPU6050, giroscopio y acelerómetro, y nos cercioramos de que funcione correctamente, entregando datos acordes a su funcionamiento.

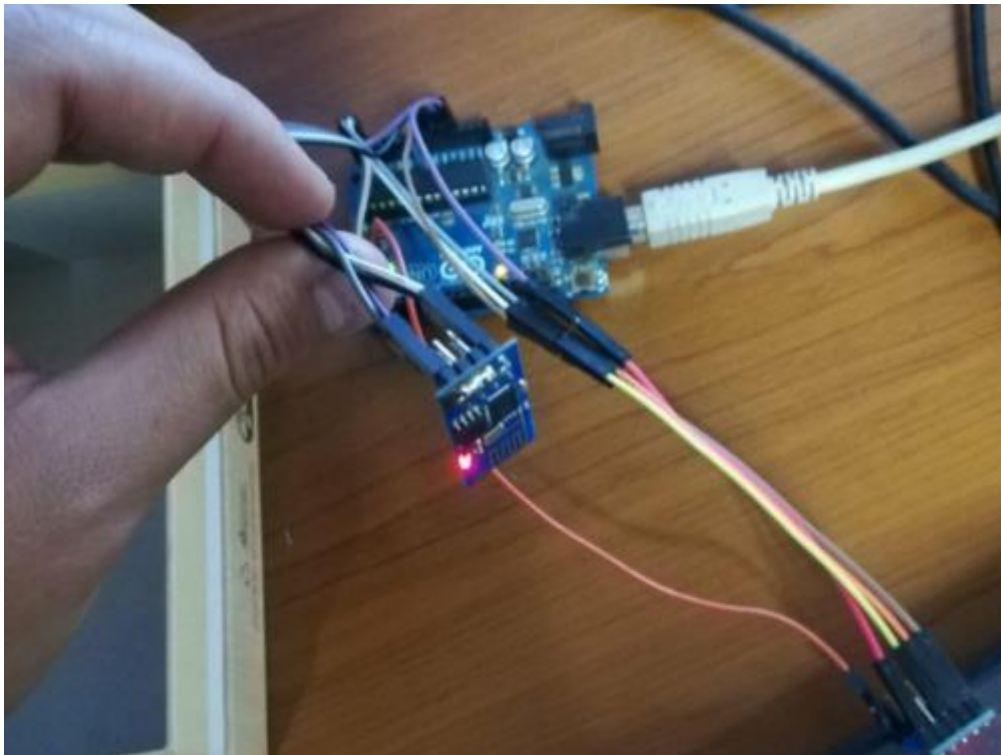


Figura 12. Arduino, MPU-6050 y ESP8266

Luego, hay que configurar el ESP8266 junto con Arduino para que no envíe los datos directamente Arduino de forma serial, si no que los envíe a través de una red Wifi.

Una vez creada la red correctamente con el microcontrolador, podemos conectarnos a esta en el puerto 23, configurando esto en el código de ejecución de ESP8266.

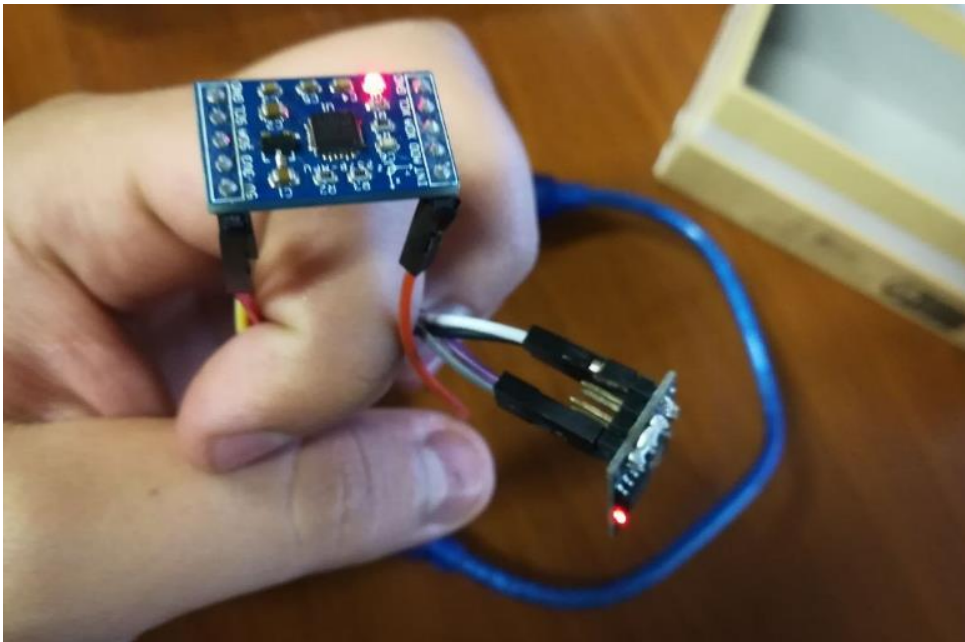


Figura 13. MPU-6050 y ESP8266 conectados.

Una vez configurado el microcontrolador ESP8266, y también corroborado el funcionamiento de nuestro sensor, conectamos los aparatos conjuntos, en nuestro Arduino r3.

Ya que nuestro microcontrolador de Wifi tiene un correcto funcionamiento, no necesitamos tener una conexión directa de nuestro Arduino a el computador, retiramos y conectamos a una batería externa todo.

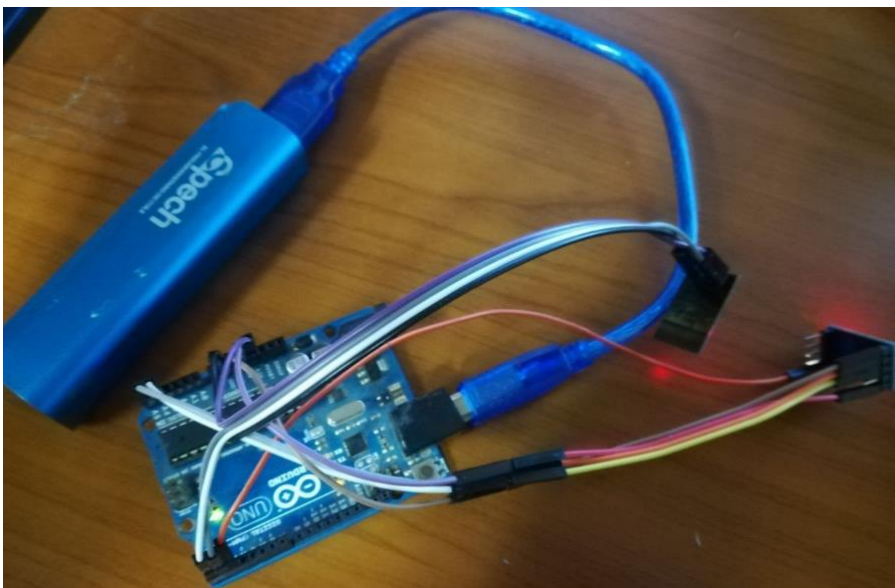


Figura14. Conexión a batería externa

Universidad del Bío – Bío Facultad de Ciencias Empresariales
Ingeniería de Ejecución en Computación e Informática

Luego introducimos cuidadosamente en una caja que contenga todos nuestros artefactos.



Figura 15. Introducción de dispositivos en caja

Luego tapamos la caja, con precaución de que ninguna de nuestras conexiones falle y nos cercioramos que funcione la captura e interpretación de datos de nuestro prototipo a la aplicación.



Figura 16. Prototipo 0.1 completo con caja y guincha

Finalmente agregamos la guincha elástica, la que nos permitirá ajustar el prototipo a nuestro brazo, muñeca o mano, para utilizarla y ver los movimientos que hagamos con él.

4 Estudio de factibilidad

El estudio de factibilidad es un instrumento que sirve para orientar la toma de decisiones en la evaluación de un proyecto y corresponde a la última fase pre-operativa o de formulación dentro de un proyecto. Esto nos permite determinar si un proyecto es viable desde distintos tipos de vista como son el técnico, económico y operativo.

A continuación, se detallará el resultado de los distintos tipos de estudios de factibilidad que se realizaron y así determinar la viabilidad del proyecto. Cada estudio nos entregará una conclusión y el conjunto de ella nos permitirá concluir con la factibilidad del proyecto general.

4.1 Factibilidad técnica

El objetivo de esta factibilidad es evaluar los equipos tecnológicos que hay en el mercado y usar dichos equipos para implementar el sistema

En este caso el proyecto parte desde cero, por lo que se debe incurrir el gasto de ellos, el desglose de los valores se especifica más abajo

- **Lenguaje y herramientas a utilizar**

El prototipo para el seguimiento de movimiento se desarrollará básicamente en un lenguaje de programación simplificado para el hardware y el bootloader ejecutado en la placa, que es el que utiliza la programación de Arduino y lenguaje processing para manipular la aplicación.

El software Arduino IDE es una plataforma abierta que facilita la programación de un microcontrolador, su principal característica es su sencillez y facilidad de uso.

- **Hardware**

Para realizar el proyecto se necesita un microcontrolador Arduino r3, sensor MPU-6050, módulo wifi ESP8266 y computador en este caso se consta con computador, los demás equipos se tienen que comprar los cuales juntos no superan un valor en el mercado de \$20.000 pesos.

En esta evaluación se usarán dos enfoques: Hardware y Software.

En cuanto a Hardware, específicamente el mini pc donde debe estar instalado el sistema propuesto, este debe cubrir con los siguientes requerimientos mínimos:

- Arduino uno r3 microcontrolador: ATmega328p
- Microcontrolador ESP8266
- Acelerómetro y giroscopio MPU 6050

- **Conclusión**

Es completamente factible el desarrollo de este prototipo, dado que se requieren ambientes de desarrollo que son de software libre y completamente gratuita, pero que, si implica la compra de los sensores y microcontroladores, pero, que no son muy costosos, y además se tienen una introducción en los conocimientos y desarrollo de aplicaciones y funcionalidades con la plataforma Arduino.

4.2 Factibilidad Económica

Recursos necesarios

A continuación, se detallarán los elementos indispensables para la realización de este proyecto:

Existe un recurso que es el computador que ya existe, por lo que no es necesario incluirlo en parte de la evaluación económica ya que no será necesario adquirirlos.

- Lenovo ideapad320, AMD A9-9420, 8 GB RAM, Windows 10, sistema operativo de 64 bits.
- Horas hombre de programador.

Evaluación Económica

- **Costos de Hardware**

Arduino uno r3:	5290
Microcontrolador ESP8266:	5000
MPU 6050:	5500
Batería externa:	3000
Costo total:	18790
Instalación:	20000
Total:	38790

- **Costos de software**

No hay, ya que los programas para el desarrollo de este proyecto son libres y completamente gratis.

Los softwares a utilizar se encuentran en los siguientes links:

<https://www.arduino.cc/en/main/software>

<https://processing.org/download/>

Estos permiten desarrollar aplicaciones para todas las personas que estén interesadas y además permite distribución sin ningún costo de licencias ni nada.

- **Costo de recursos humanos**

Los costos de recursos humanos son estimaciones debido a que el proyecto de título es completamente gratis, pero para el correcto estudio de factibilidad se detallara a continuación.

Horas hombres= HH.

HH en análisis y diseño:	300 HH (60 días media jornada)
HH en construcción:	150 HH (30 días media jornada)
HH diseño final:	16 HH (2 días jornada completa)

Total, de HH: 466 HH

Valor HH programador analista: 2.500(aproximadamente)

Costo total del proyecto en pesos: \$1.165.000

- **Conclusión**

Es económicamente posible desarrollar un prototipo como este, puesto que no se considera el costo de un computador necesario para el desarrollo del proyecto ya que se dispone del. Los elementos para la construcción y posterior desarrollo son relativamente baratos, por lo que no es necesario realizar grandes gastos, contando con los recursos computacionales y humanos.

4.3 Factibilidad operativa

Este proyecto, es un prototipo que se utilizara de base para un proyecto que consta de futuros clientes. Por tanto, está desarrollado para un futuro cliente. Quienes solo deberán saber cómo ubicar los sensores, para luego leer y traducir los movimientos de las personas en una específica especialidad (kinesiología).

Este proyecto fue presentado delante de una especialista quien será la futura cliente a usar este prototipo quien se mostró entusiasta de participar y apoyar la construcción de prototipos de seguimiento en tiempo real y a un bajísimo costo a comparación de las herramientas actuales.

La especialista tiene claro que no reemplazara a los sistemas actuales, pero si puede facilitar o favorecer la tarea que ellos a menudo realizan, ya sea con tiempo o simplicidad para algunos casos, existe un deseo e interés de colaborar y participar en el proyecto.

4.4 Conclusión de la factibilidad

Demostrado en el estudio de factibilidad antes mencionado, tanto la factibilidad técnica, económica y operativa se observan resultados favorables para la viabilidad del proyecto. El proyecto servirá como prototipo para dar un comienzo a una investigación más grande, en la empresa SmartGrows, además de no conllevar un gasto muy grande o excesivo para un prototipo.

5 CONCLUSIONES

En el proyecto desarrollado en el presente informe se logró diseñar e implementar un prototipo inalámbrico que muestra los movimientos en tiempo real, además de cumplir con los objetivos planteados en el desarrollo, ya sea general y específicos, también se logró mejorar el prototipo con mejoras que surgieron en la construcción. El resultado del prototipo e investigación realizados, dieron por consecuencia un final de proyecto totalmente alentador y viable para un sistema de seguimientos en tiempo real e inalámbricos. Este a su vez, da la oportunidad y claridad de opción para seguir con un desarrollo futuro. Mejoras tales como prototipo más pequeño, prototipo con una mayor duración de batería o mayor cantidad de prototipos interconectados para una interpretación más específica de movimientos como la de las articulaciones.

El desarrollo de este proyecto, se tuvo que realizar de manera ágil, con disponibilidades de horarios de la empresa, y un poco de malos entendidos con tiempos y entregas, por problemas externos, pero los cuales se supieron sobrellevar para lograr un desarrollo completo y fructífero. Se tuvo que investigar bastante ya que los conocimientos de los módulos y software utilizados, la mayoría, no se tenía ningún conocimiento previo.

En el contexto profesional, se utilizaron muchos de los conocimientos adquiridos durante todos los años de la carrera, Ya sea en conocer conceptos como en el incentivo de investigación y mejoras de un proyecto, sobre todo en el área de Arduino y programación, creo que el desarrollar y ver resultados de inmediatos ya sea en un prototipo o una simple luz, me incentivaron a hacer este proyecto, en el ámbito de la comprensión de problemas me sirvió mucho la complejidad de mi carrera, para siempre planear respuestas o salidas para un problema propuesto, sin lugar a dudas este proyecto y cada etapa superada en mi carrera han sido oportunidades para seguir en el camino constante superación, las cuales me han incentivado y ayudado en la formación como futuro profesional.

6 BIBLIOGRAFIA DIGITAL

A continuación, citare y hare referencia a los datos que me permitieron generar el proyecto, cabe señalar que todas las referencias citadas son y fueron utilizadas en formato digital.

<https://www.arduino.cc/>

Librería principal de Arduino enseñada en el ramo electivo de Arduino.

<https://sites.google.com/site/modelodeprototipo/etapas-para-la-elaboracion-del-modelo-de-prototipos>

Modelo para la preparación de un prototipo.

<https://programarfacil.com/podcast/esp8266-wifi-coste-arduino>

Programación de módulo ESP8266.

http://www.naylampmechatronics.com/blog/45_Tutorial-MPU6050-Aceler%C3%B3metro-y-Giroscopio.html

Introducción y explicación de uso de sensor MPU-6050.

<http://diymakers.es/arduino-processing-primeros-pasos/>

Primeros pasos en processing programando con Arduino.

<https://github.com/technicolorenyv/Processing-Libraries/tree/master/toxiclibscore>

Librerías esenciales para el uso de processing junto con Arduino.

<https://github.com/jarzebski/Arduino-MPU6050>

Librería para usar sensor MPU6050.

http://www.academia.edu/14250135/Linguistic_description_of_the_human_gait_quality

Investigación que me incentivo.

7 GLOSARIO

A continuación, hare una recopilación de definiciones y explicaciones de palabras claves y utilizadas en el presente informe.

1. TCP/IP: Son siglas de protocolo de control de transmisión de internet.
2. I2C: Es un protocolo de comunicación serial.
3. Arduino: Es una plataforma de prototipos electrónica de código abierto.
4. MPU-6050: Es un sensor y acelerómetro para Arduino con 6 grados de libertad.
5. ESP-8266: Es un chip Wi-Fi de bajo coste con la pila TCP/IP completa y capacidad de microcontrolador.
6. Processing: Es un lenguaje de programación y entorno de desarrollo integrado de código abierto basado en java.
7. Prototipo: Es un primer ejemplar que se fabrica de una figura, un invento u otra cosa.
8. IMU: Unidad de medida inercial.
9. Sensor: Dispositivo que capta magnitudes físicas.
10. Microcontrolador: Es un circuito integrado programable, capaz de ejecutar ordenes grabadas en su memoria.
11. Atmega328: Es un circuito integrado de alto rendimiento basado en microcontrolador RISC.
12. Bootloader: Es el nombre en inglés para el gestor de arranque de un dispositivo.
13. Latencia: Es la suma de retardos temporales dentro de una red.
14. Software: Es el soporte lógico de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios para la realización de tareas específicas.
15. Hardware: Hace referencia a las partes físicas tangibles de un sistema informático, como por ejemplo sus componentes eléctricos, electrónicos, mecánicos etc.
16. Bit: es el acrónimo de dígito binario.