

Universidad Tecnológica de Cancún

ORGANISMO PUBLICO DESCENTRALIZADO DEL GOBIERNO DEL ESTADO DE QUINTANA ROO

* **“SIT-PROVIDER”.**
* MEMORIA
* Que para obtener el Título de:
* **Ingeniería en Tecnologías de la Información y Comunicación**
* **P R E S E N T A :**
* **José Luis Ortiz Acosta**
* **11392317**
* **Asesor académico: MTI. Irving Tolosa Garma.**
* **Empresa: Universidad Tecnológica de Cancún.**

**Asesor empresarial: Ing. Ricardo Delgado Arenal.**

Cancún Quintana Roo Abril de 2015

DATOS GENERALES DEL ALUMNO

Nombre del alumno: **Jose Luis Ortiz Acosta**

Matrícula: **11392317**

Generación: **2013 - 2015**

Fecha de inicio y de terminación de la estadía: **13 de Enero a 2 de Abril de 2015**

Dirección: **C 4TA priv kabah Smz 248 mz 22 lot 1 #37**

Teléfono: **9981886519**

Asesor académico: **MTI. Irving Tolosa Garma.**

AUTORIZACIÓN DE DIGITALIZACIÓN

AUTORIZACIÓN DE ASESORES

**ÍNDICE**

[DATOS GENERALES DEL ALUMNO 3](#_Toc414179222)

[AUTORIZACIÓN DE DIGITALIZACIÓN 4](#_Toc414179223)

[AUTORIZACIÓN DE ASESORES 5](#_Toc414179224)

[PARTE 2. CONTEXTO LABORAL 8](#_Toc414179225)

[Datos Generales de la Empresa 8](#_Toc414179226)

[ANTECEDENTES HISTÓRICOS DE LA EMPRESA 9](#_Toc414179227)

[DESCRIPCIONES DE LA EMPRESA 10](#_Toc414179228)

[Estructura orgánica. 12](#_Toc414179229)

[DESCRIPCIÓN DEL ÁREA DONDE SE REALIZARÁN LAS PRÁCTICAS PROFESIONALES 13](#_Toc414179230)

[PARTE 3. CONTEXTO ACADÉMICO PROFESIONAL 14](#_Toc414179231)

[I. INTRODUCCIÓN 14](#_Toc414179232)

[II. JUSTIFICACIÓN 15](#_Toc414179233)

[III. OBJETIVOS DEL PROYECTO 16](#_Toc414179234)

[IV. FUNDAMENTOS 17](#_Toc414179235)

[Sistemas Embebidos. 17](#_Toc414179236)

[Arduino. 18](#_Toc414179237)

[Arduino Uno-R3. 19](#_Toc414179238)

[Ethernet Shield. 20](#_Toc414179239)

[Actuadores. 21](#_Toc414179240)

[Servomotor. 22](#_Toc414179241)

[Sensor. 23](#_Toc414179242)

[Botones. 24](#_Toc414179243)

[V. DESARROLLO DEL PROYECTO 25](#_Toc414179244)

[V.1 Cronograma de actividades 25](#_Toc414179245)

[V.2 Herramientas de análisis 25](#_Toc414179246)

[V.3 Herramientas de desarrollo 34](#_Toc414179247)

[V.4 Implementación del proyecto 40](#_Toc414179248)

[V.5 Análisis de resultados. 47](#_Toc414179249)

[V.6 Estudio de factibilidad 50](#_Toc414179250)

[VI. CONCLUSIONES Y CONTRIBUCIÓN TÉCNICA 51](#_Toc414179251)

[BIBLIOGRAFÍA Y CONSULTAS REALIZADAS 53](#_Toc414179252)

[ANEXOS 53](#_Toc414179253)

[Anexo 1. TABLA DE FIGURAS O ILUSTRACIONES 53](#_Toc414179254)

[GLOSARIO 56](#_Toc414179255)

PARTE 2. CONTEXTO LABORAL

Datos Generales de la Empresa

**Nombre de la empresa:** SITWIFI SA de CV.

**Dirección:** Av. Bonampak, CP 77504, Orbis Coworking Oficina 18

**Sector y giro empresarial:** Tecnológico y de Soporte.

**Servicios que ofrecen:**

* Disponibilidad de infraestructura Service Level Agreement.
* Network Operations Center.
* Sistemas de tickets en línea.
* Garantía de cobertura.
* Garantías extendidas de infraestructura.
* Reportes mensuales.
* Resolución de problemas en sitio y remoto.
* Mantenimiento.

**Nombre del Proyecto:** SIT-PROVIDER

**Asesor de la empresa:** Ing. Ricardo Delgado Arenal.

**Cargo del Asesor Empresarial:** Coordinador de soporte.

ANTECEDENTES HISTÓRICOS DE LA EMPRESA

SIT-WIFI es una empresa enfocada a dar soluciones integrales de acceso inalámbrico a internet. Implementamos soluciones a la medida para cada cliente, tomando en cuenta sus necesidades técnicas y financieras.

A través de los estudios de cobertura y demanda que llevamos a cabo en las instalaciones de cada cliente, diseñamos la infraestructura de red para cubrir sus necesidades específicas, recomendado los equipos adecuados para su óptima operación.

En nuestras soluciones no se requiere de inversión inicial por parte del cliente, debido a que nuestro modelo es conocido como infraestructura de Servicio y puesta en marcha en toda la infraestructura de red.

SITWIFI al ser el propietario de la infraestructura tiene la responsabilidad de garantizar, dar mantenimiento, operación y administración a la infraestructura, garantizando el nivel de servicio requerido.



DESCRIPCIONES DE LA EMPRESA

**Misión**

Nuestra misión es solucionar con calidad y tecnología de vanguardia, las necesidades de acceso inalámbrico de internet de nuestros clientes, sin necesidad de inversión alguna.

**Visión**

Garantizar la plena satisfacción de nuestros clientes, ofreciendo en el diseño de la solución de internet productos de la mejor calidad y precio adecuado bajo un esquema de corresponsabilidad entre proveedores, colaboradores e instancias de participación a través de un costumer service.

**Política Salarial.**

Crear un sistema de enumeración ordenado, equitativo para la organización y para los empleados, que motive eficazmente el trabajo productivo y el cumplimiento de los objetivos y metas de la compañía.

**Evaluación de desempeño**.

Las elevaciones salariales se realizaran basándose en la evaluación de desempeño y la situación financiera de la empresa, se exceptúan las disposiciones gubernamentales.

Las evaluaciones de desempeño se harán semestralmente en Julio y en Enero. En julio se revisara el avance de las metas establecidas a principio de año y en Enero se hara la evaluación final. Esta evaluación determinara si el colaborador es acreedor a un aumento de sueldo, a algún bono, promoción o baja de la empresa.

**Políticas de vacaciones**.

La administración deberá planificar las vacaciones de sus trabajadores, con la finalidad de propiciar su necesario descanso anual, asegurando que la ausencia temporal de aquellos no cause inconvenientes al normal desarrollo de actividades.

**Política de selección y contratación de personal**.

Establecer las normas aplicables a las actividades de selección y contratación del recurso humano que permitan escoger personas idóneas, que se ajusten a los requerimientos de la compañía y a los perfiles establecidos para alcanzar sus objetivos estratégicos y asegurar su futuro desarrollo.

**Política de horarios laborales y asistencia**.

Todos los colaboradores tienen la obligación de iniciar puntualmente sus labores, de acuerdo al horario establecido, registrar su asistencia diaria y cuando se realicen capacitaciones y/o reuniones.

La asistencia a las reuniones de trabajo es de manera obligatoria. De presentarse algún inconveniente, el trabajador deberá comunicar a su superior.

**Política de vestimenta**.

Todo colaborador deberá de usar el uniforme de la empresa, en caso de alguna contingencia deberá de vestir de manera formal.

El día viernes no es obligatoria el uniforme, se podrá ir de jeans usando la camisa de vestir. En caso de tener alguna cita programada se tendrá que usar el uniforme o estilo formal.

Se prohíbe el uso de tenis, camiseta o cualquier vestimenta con logotipos ofensivos.

**Política de Seguridad de Información.**

Proteger la información estratégica de la compañía de la misma manera la cartera de clientes y normar sus niveles de acceso y confidencialidad. Todos los empleados, deberán firmar un acuerdo de confidencialidad.

Estructura Organizacional

Director

Operaciones Sureste

Gerente

Comercial

Gerente

Costumer Service

IT

Master

Gerente

Soporte Técnico

IT

Concierge

Monitoreo

Técnicos

Ejecutivo

Ventas

Ejecutivo

WIFIMEDIA

Administrador

Cuentas

Arquitecto

Red

Ilustración Organigrama de SITWIFI Sureste

DESCRIPCIÓN DEL ÁREA DONDE SE REALIZARÁN LAS PRÁCTICAS PROFESIONALES

**Nombre del área:** Oficina Orbis #18

**Objetivos:** Los alumnos comprendan el análisis, la planeación y la puesta en marcha de programas de mantenimiento de los equipos Ruckus, a fin de adquirir los conocimientos para realizar la planeación de redes inalámbricas, disminuir los costos y elevar la rentabilidad de la empresa.

**Funciones:**  
En el ejercicio de sus funciones técnicas:

Colaborar con la dirección o gerencia y coordinar su actividad con los demás puestos técnicos para el cumplimiento de los programas de producción dentro de los montos, calidades y tiempos que se establezcan.

Conduce y supervisa las operaciones de mantenimiento, la seguridad técnica y el buen funcionamiento de los equipos.

Las funciones de Ingeniería en Tecnologías de la información y comunicación, pueden ser desempeñadas en empresas del sector industrial y hotelero de diferente tamaño y organización:

En empresas de gran tamaño, independientemente de que el técnico figure o no al frente de un departamento específico, está capacitado para asesorar, gestionar y ejecutar las decisiones de la dirección o la gerencia.

En las empresas medianas, donde la división del trabajo tiene menos segmentos, puede cumplir las mismas funciones que desempeña en la gran empresa, pero con mayor nivel de integración, mayor autonomía y capacidad de decisión.

Campo de acción profesional: Compañías de servicio de hotelero, soporte y las empresas del sector industrial.

PARTE 3. CONTEXTO ACADÉMICO PROFESIONAL

1. INTRODUCCIÓN

El proyecto es desarrollado para la empresa SIT-WIFI y tiene como objetivo diseñar e implementar el proyecto SIT-PROVIDER.

Este proyecto se desarrollara en 3 etapas, planificación, el diseño y codificación

En la etapa de planificación se realizara el análisis de requerimientos, las herramientas de desarrollo a utilizar las cuales son: Photoshop, Sublime Text, Lovely Chart, UML, Workbench, PHP, MySQL, HTML5, AJAX.

En la etapa de diseño se realizara la maquetación de las vistas y estructura de las mismas para facilidad de uso del usuario final.

Para la etapa de codificación de SIT-PROVIDER se usará un editor de texto llamado Sublime Text para crearla y para darle diseño se empleará CSS ya que es una herramienta fácil de manejar. La aplicación será sencilla para que el usuario no tenga problemas para operarla.

Por consiguiente se empleará el lenguaje PHP el cual se comunicará con la aplicación, en conjunto con AJAX para un flujo eficaz de la aplicación.

Las ventajas que ofrece este proyecto, es la facilidad y libertad que propone el hecho que este desarrollado vía web ya que cualquier dispositivo que cuente con un programa para explorar el internet lo pueda usar, cualquier persona que se encuentre del otro lado del mundo pueda manejar la aplicación sin necesidad de estar cerca.

1. JUSTIFICACIÓN

El objetivo principal de este proyecto parte de la necesidad de aprovechar todo aquel recurso con el que cuente la empresa, para obtener un mayor control de los productos almacenados en bodega, Enviados a garantía y aquellos que se encuentren en producción, contemplando que sea posible verificar dicha información, sin importar que el usuario se encuentre en sitio.

Para que esto sea posible es necesario implementar herramientas de vanguardia como lo es HTML5 + CSS3 en conjunto de PHP y MySQL que son plataformas de software libre que permite reducir los costos del proyecto, lo cual hace que sea una parte fundamental del proyecto al permitir distribuir los recursos, ya que la aplicación va estar alojada en un servidor local con enlace a internet que permitirá el acceso desde cualquier punto que se realiza la consulta. Otra de las ventajas es que al ser una aplicación de tipo web es posible contar con la suficiente documentación necesaria y partir de Frameworks de trabajo que permiten eficientar el código a través de la estandarización de los desarrollos hoy en día existentes en el mercado, además de permitir una flexibilidad en la parte de la codificación de este proyecto al implementar las marcos de trabajo recomendados y las mejores prácticas.

Los beneficios que se podrán obtener de este proyecto son de alto valor para la empresa debido a que la empresa no cuenta con algo semejante ya sea de tipo Open Source o bajo Licenciamiento, tomando en cuenta que es una aplicación desarrollada a la medida de las necesidades de la empresa y sin olvidar que al ser de tipo web permitirá ser consultada remotamente sin importar que tipo de dispositivo realice la solicitud, ya que estará bajo el esquema responsive.

1. OBJETIVOS DEL PROYECTO

Objetivo general

Diseñar e implementar una aplicación web para ofrecer un control de inventarios así como consultas de ubicación de equipos utilizando software avanzado y a la vanguardia con tecnología moderna.

Objetivos específicos

* Garantizar mediante la programación embebida los mecanismos y la estabilidad del robot.
* Crear una interfaz amigable para el usuario.
* Utilizar el micro controlador como servidor de la página web para la interacción manual con el robot.
* Programar el microcontrolador para un funcionamiento óptimo de la Interfaz WEB y del robot en conjunto.

1. FUNDAMENTOS

Sistemas Embebidos.

Un sistema embebido o empotrado es un sistema de computación diseñado para realizar una o algunas pocas funciones dedicadas frecuentemente en un sistema de computación en tiempo real. Al contrario de lo que ocurre con los ordenadores de propósito general (como por ejemplo una computadora personal o PC) que están diseñados para cubrir un amplio rango de necesidades, los sistemas embebidos se diseñan para cubrir necesidades específicas. En un sistema embebido la mayoría de los componentes se encuentran incluidos en la placa base y muchas veces los dispositivos resultantes no tienen el aspecto de lo que se suele asociar a una computadora. Algunos ejemplos de sistemas embebidos podrían ser dispositivos como un taxímetro, un sistema de control de acceso, la electrónica que controla una máquina expendedora o el sistema de control de una fotocopiadora entre otras múltiples aplicaciones. (Barr, 2007)

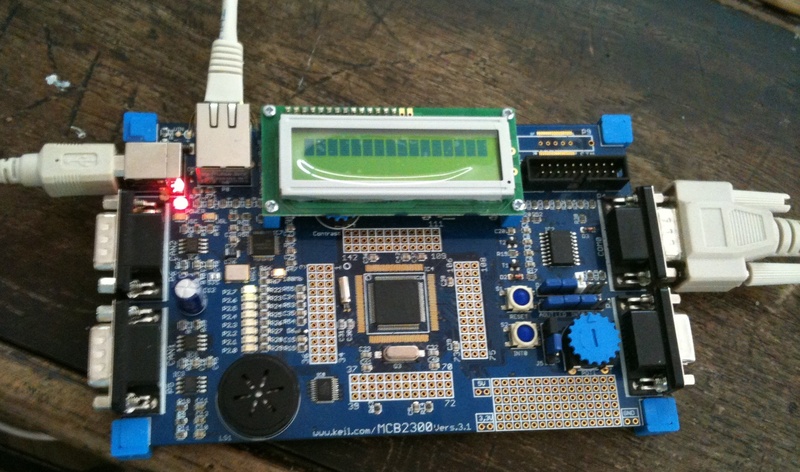
****Por lo general los sistemas embebidos se pueden programar directamente en el lenguaje ensamblador del microcontrolador o microprocesador incorporado sobre el mismo, o también, utilizando los compiladores específicos, pueden utilizarse lenguajes como C o C++; en algunos casos, cuando el tiempo de respuesta de la aplicación no es un factor crítico, también pueden usarse lenguajes interpretados como JAVA.

Ilustración . Sistema embebido

Puesto que los sistemas embebidos se pueden fabricar por decenas de millares o por millones de unidades, una de las principales preocupaciones es reducir los costes. Los sistemas embebidos suelen usar un procesador relativamente pequeño y una memoria pequeña para ello.

Arduino.

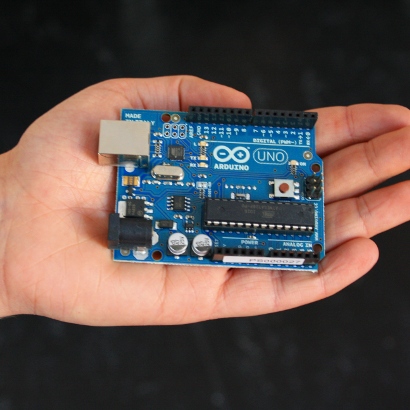
Arduino es una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles y fáciles de usar. Se creó para artistas, diseñadores, aficionados y cualquiera interesado en crear entornos u objetos interactivos.

Ilustración . Placa Arduino

Arduino puede tomar información del entorno a través de sus pines de entrada de toda una gama de sensores y puede afectar aquello que le rodea controlando luces, motores y otros actuadores. El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en *Wiring*) y el entorno de desarrollo Arduino (basado en *Processing*). Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a un ordenador, si bien tienen la posibilidad de hacerlo y comunicar con diferentes tipos de software. (Arduino, 2005)

Esto es posible debido a que Arduino se comunica mediante la transmisión de datos en formato serie que es algo que la mayoría de los lenguajes anteriormente citados soportan. Para los que no soportan el formato serie de forma nativa, es posible utilizar software intermediario que traduzca los mensajes enviados por ambas partes para permitir una comunicación fluida. Es bastante interesante tener la posibilidad de interactuar Arduino mediante esta gran variedad de sistemas y lenguajes puesto que dependiendo de cuales sean las necesidades del problema que vamos a resolver podremos aprovecharnos de la gran compatibilidad de comunicación que ofrece.

Arduino está basado en C y soporta todas las funciones del estándar C y algunas de C++.

Arduino Uno-R3.

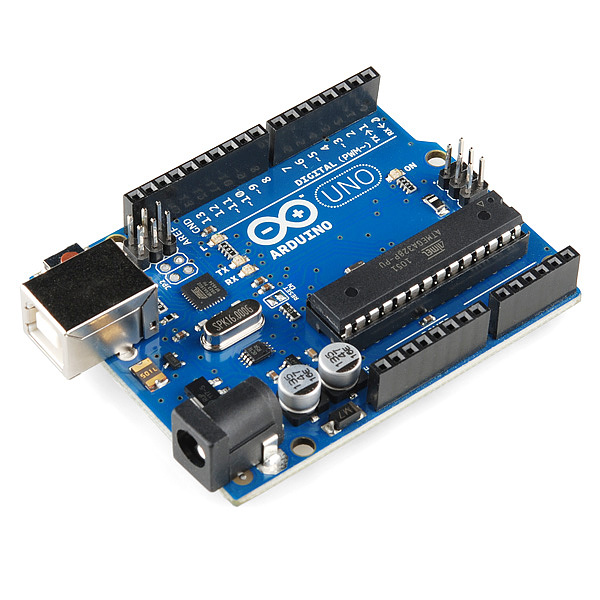
Esta es la más nueva versión de Arduino UNO: R3. Esta tarjeta incorpora nuevas características como el empleo de un ATmega16U2 en lugar del 8U2 usado en versiones anteriores de Arduino (o el FTDI de generaciones previas). Esto permite tasas de transferencia más altas y brinda más memoria. No se requieren drivers para instalarlo bajo Linux o Mac (La versión de Windows incluye los drivers). Con el 16U2 te permite que el UNO sea reconocido como un teclado, mouse, joystick o cualquier otro periférico USB. (Arduino, 2005)

Ilustración . Placa Arduino Uno

El Uno R3 también incorpora nuevos pines como: SDA, SCL al lado de AREF. Así mismo se presenta un par de nuevos pines cerca del pin de RESET. Uno de ellos es el IOREF que permite que los shields se adapten al voltaje empleado por la tarjeta; el otro pin está reservado para futuras versiones. El Arduino UNO R3 se puede emplear con todos los shields existentes así como con las versiones más nuevas que ya hacen uso de estos pines.

Características:

* Microcontrolador Atmega 328 @ 16MHz.
* Voltaje de trabajo 5v.
* Tensión de alimentación 7v- 12v.
* Conexión USB.
* 16 Entradas/salidas digitales (DIO).
* 6 Entradas analógicas.
* Corriente máxima de 40mA.

Ethernet Shield.

Este shield te permite nada más y nada menos que conectar tu Arduino a internet. Está basado en el chip de Ethernet Wiznet W5100 con funcionalidades de IP tanto para TCP como UDP. El Arduino Ethernet Shield soporta hasta 4 conexiones simultáneas. Utiliza la librería Ethernet para escribir rápidamente programas que te conecten a internet empleando este shield. (Arduino, 2005)

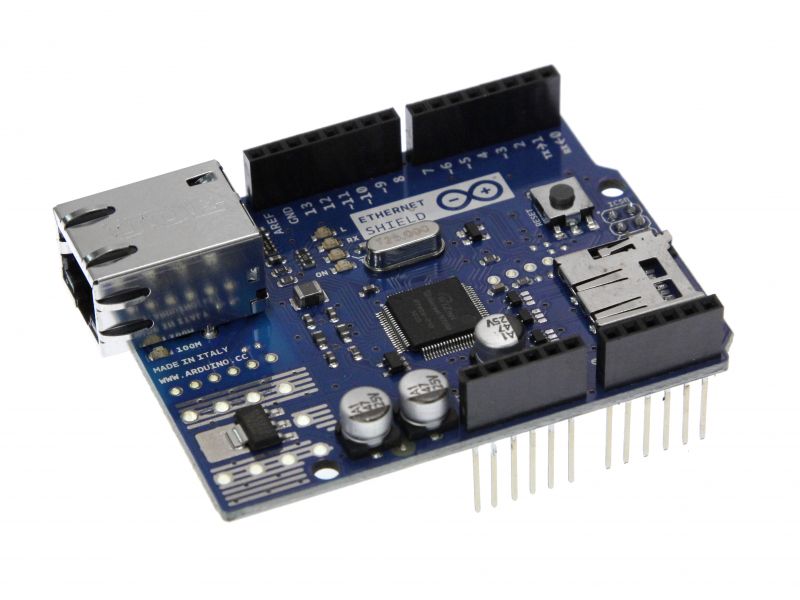
Una gran ventaja de este shield es que es apilable por lo que se puede disponer de todos los pines en otros shields.

Ilustración . Ethernet Shield

Esta es la última versión de este shield que incluye un slot para tarjetas micro-SD, el cual puede ser empleado para almacenar archivos que podrás poner disponibles a través de la red. Es compatible con el Arduino UNO y el Mega (empleando la librería que viene en la versión de software Arduino 0019). La librería para el manejo de la tarjeta todavía no está incluida en la distribución estándar de Arduino pero se puede emplear la librería desarrollada por Bill Greiman.

En esta revisión del shield también se incluye un controlador de reset, esto es para asegurarse que el módulo W5100 Ethernet inicie correctamente al conectarlo a la electricidad. Revisiones previas de este shield no eran compatibles con el Arduino Mega y necesitaban reset manual después de conectarlo. (SparkFun, 2008)

Para usar la Ethernet Shield solo hay que montarla sobre la placa Arduino. Para cargar los sketches a la placa conectarla al ordenador mediante el cable USB como se hace normalmente. Una vez que el sketch ha sido cargado se puede desconectar la placa del ordenador y alimentarla desde una fuente externa.

Actuadores.

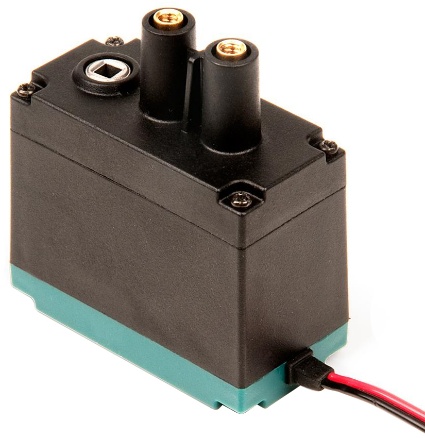
Un actuador es un dispositivo mecánico que está diseñado para convertir una señal eléctrica en una acción física. Se suelen utilizar en la fabricación de aplicaciones industriales y puede ser utilizada para válvulas de control de flujo, bombas, unidades de posicionamiento, motor, interruptor, relés y medidores. Los actuadores están disponibles en diferentes formas, tales como actuador neumático, actuador giratorio, el actuador lineal, actuador eléctrico, actuador de pistón y los actuadores hidráulicos. Cada forma es operada por una fuente específica de energía, por ejemplo, la energía mecánica, eléctrica y potencial, almacenada en líquidos o gases comprimidos.

Ilustración . Motor

Existen varios tipos de actuadores como son:

* Electrónicos
* Hidráulicos
* Neumáticos
* Eléctricos

Un actuador neumático se utiliza normalmente para comprimir el aire como una fuente de energía que entonces se convierte en movimiento. Cada tipo de actuador produce un movimiento oscilatorio o específica como lineal, dependiendo del tipo del actuador. Los actuadores neumáticos contienen un pistón, un cilindro y las válvulas o puertos. El diafragma cerrado protege el pistón y mantiene el aire en la parte superior del cilindro. Esta construcción permite que el aire a presión para empujar el diafragma hacia abajo, moviendo los debajo del pistón, que está conectado a las partes internas del actuador. Pocos ejemplos de actuadores neumáticos son: pinzas, cilindros de unión de carreteras, actuadores rotativos, generadores de vacío, etc. para el funcionamiento eficaz de los actuadores, de mayor tamaño de un pistón se recomienda debido a un pistón más grande sería más eficaz si el suministro de aire es baja, lo que permite la misma fuerza con una cantidad menor de entrada. (Kosow, 2011)

Ilustración . Actuador Neumático

Servomotor.

Un servomotor (también llamado servo) es un dispositivo similar a un motor de corriente continua que tiene la capacidad de ubicarse en cualquier posición dentro de su rango de operación, y mantenerse estable en dicha posición.1

Un servomotor es un motor eléctrico que consta con la capacidad de ser controlado, tanto en velocidad como en posición.

Los servos se utilizan frecuentemente en sistemas de radio control y en robótica, pero su uso no está limitado a éstos. Es posible modificar un servomotor para obtener un motor de corriente continua que, si bien ya no tiene la capacidad de control del servo, conserva la fuerza, velocidad y baja inercia que caracteriza a estos dispositivos.

Ilustración . ServoMotor

Está conformado por un motor, una caja reductora y un circuito de control. También potencia proporcional para cargas mecánicas. Un servo, por consiguiente, tiene un consumo de energía reducido.

La corriente que requiere depende del tamaño del servo. Normalmente el fabricante indica cuál es la corriente que consume. La corriente depende principalmente del par, y puede exceder un amperio si el servo está enclavado.

En otras palabras, un servomotor es un motor especial al que se ha añadido un sistema de control (tarjeta electrónica), un potenciómetro y un conjunto de engranajes. Con anterioridad los servomotores no permitían que el motor girara 360 grados, solo aproximadamente 180; sin embargo, hoy en día existen servomotores en los que puede ser controlada su posición y velocidad en los 360 grados. Los servomotores son comúnmente usados en modelismo como aviones, barcos, helicópteros y trenes para controlar de manera eficaz los sistemas motores y los de dirección. (Hangar, 2007)

Sensor.

Un sensor es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, etc. Una magnitud eléctrica puede ser una resistencia eléctrica (como en una RTD), una capacidad eléctrica (como en un sensor de humedad), una Tensión eléctrica (como en un termopar), una corriente eléctrica (como en un fototransistor), etc.

Un sensor se diferencia de un transductor en que el sensor está siempre en contacto con la variable de instrumentación con lo que puede decirse también que es un dispositivo que aprovecha una de sus propiedades con el fin de adaptar la señal que mide para que la pueda interpretar otro dispositivo. Como por ejemplo el termómetro de mercurio que aprovecha la propiedad que posee el mercurio de dilatarse o contraerse por la acción de la temperatura. Un sensor también puede decirse que es un dispositivo que convierte una forma de energía en otra. (Kevan, 2013)

Ilustración . Sensores

Características:

* Rango de medida.
* Precisión.
* Linealidad
* Rapidez de respuesta.

Botones.

Un botón o pulsador es un dispositivo utilizado para activar alguna función. Los botones son de diversa forma y tamaño y se encuentran en todo tipo de dispositivos, aunque principalmente en aparatos eléctricos o electrónicos. Los botones son por lo general activados al ser pulsados, normalmente con un dedo. Corriente mientras es accionado. Cuando ya no se actúa sobre él vuelve a su posición de reposo.

Puede ser el contacto normalmente cerrado en reposo NC, o con el contacto normalmente abierto en reposo NA.

Consta del botón pulsador; una lámina conductora que establece contacto con los dos terminales al oprimir el botón, y un muelle que hace recobrar a la lámina su posición primitiva al cesar la presión sobre el botón pulsador.

El botón de un dispositivo electrónico funciona por lo general como un interruptor eléctrico, es decir en su interior tiene dos contactos, al ser pulsado uno, se activará la función inversa de la que en ese momento este realizando, si es un dispositivo NA (normalmente abierto) será cerrado, si es un dispositivo NC (normalmente cerrado) será abierto.

Ilustración . Dispositivo NC

1. DESARROLLO DEL PROYECTO

V.1 Cronograma de actividades

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Cronograma de actividades** | | | | | | | | | | | | | | | | | |
| Nombre del proyecto | | Robot de 2 grados de libertad controlado vía WEB con programación embebida. | | | | | | | | | | | | | | | |
| Periodo de la realización | | **13 de mayo a 9 de agosto de 2013.** | | | | | | | | | | | | | | | |
| **N°** | **Actividades /Producto** | **MAYO** | | | | **JUNIO** | | | | **JULIO** | | | | **AGOSTO** | | | |
| **9** | **16** | **23** | **30** | **6** | **13** | **20** | **27** | **4** | **11** | **18** | **25** | **1** | **2** |  |  |
| 1 | Inicio del proyecto |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Planeación del proyecto |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Revisión de actividades |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3.1 Diseño de tarjeta electrónica |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3.2 Pruebas e implementación. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Ejecución del proyecto |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 4.1 Diseño de Página Web. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 4.2 Programación del Microcontrolador. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 4.3 Pruebas de caja negra y Blanca. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 4.4 Elaboración de tarjeta. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 4.5 Ensamble del Robot |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Control del proyecto |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Documentación del proyecto |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

V.2 Herramientas de análisis

* 1. **Modelo de ciclo de vida**

Modelo de Prototipos.

Consiste en que el prototipo debe ser construido en poco tiempo, usando los programas adecuados y no se deben utilizar muchos recursos.

El diseño rápido se centra en aquellos aspectos de software que serán visibles para el usuario final. Este diseño conduce a un prototipo que es evaluado por el cliente para una retroalimentación, gracias a esta se irá refinando el sistema a desarrollar. La interacción ocurre cuando el prototipo se ajusta para satisfacer las necesidades del cliente, permitiendo al desarrollador entender lo que se debe hacer y al cliente ver resultados a corto plazo.

Cabe mencionar que este modelo es útil cuando el cliente conoce los objetivos generales para el software, pero no identifica los requisitos detallados de entrada, procesamiento o salida

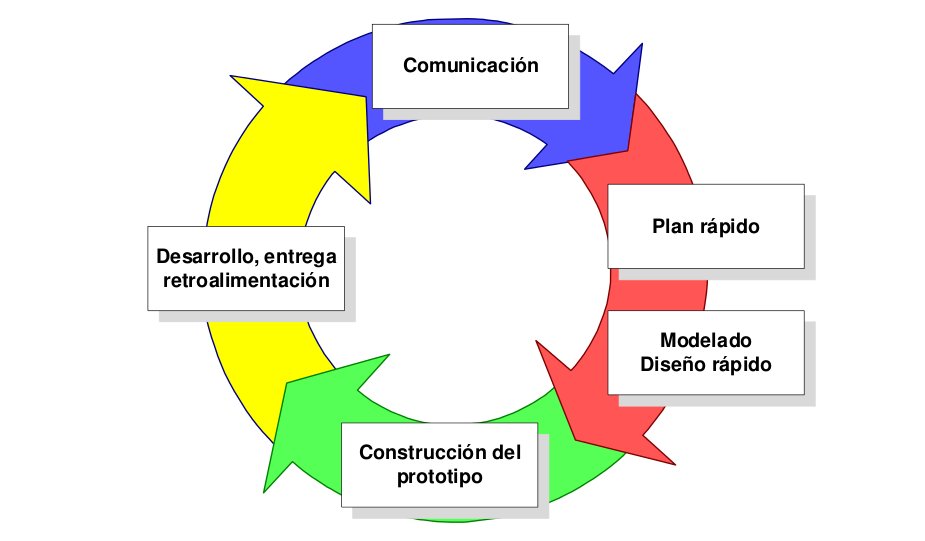


Ilustración . Etapas del modelo de prototipos

ETAPAS

Las etapas del modelado por prototipos (figura 21) son:

* Plan rápido
* Modelado, diseño rápido
* Construcción del Prototipo
* Desarrollo, entrega y retroalimentación
* Comunicación

El modelo por prototipos encaja de manera correcta en el desarrollo del proyecto a realizar.

Por experiencia en cada prototipo habrá errores y detalles que corregir por lo tanto el primer producto sabremos que no será el definitivo.

Los clientes potenciales de este producto conocerán los objetivos del proyecto, pero al desconocer sobre el desarrollo de mecanismos, este modelado es precisamente el indicado para poder mientras se avanza integrar y envolver al cliente con lo que se está realizando.

Al tener un plazo corto de tiempo para realizar y documentar el proyecto, este modelado ayuda en el sentido de que aunque el primer prototipo no será el definitivo, tendrá la esencia completa de lo que se pretende desarrollar, por lo tanto se verán resultados tajantes y concisos.

* 1. **Diagramas de Casos de Uso y sus cuadros de descripción**

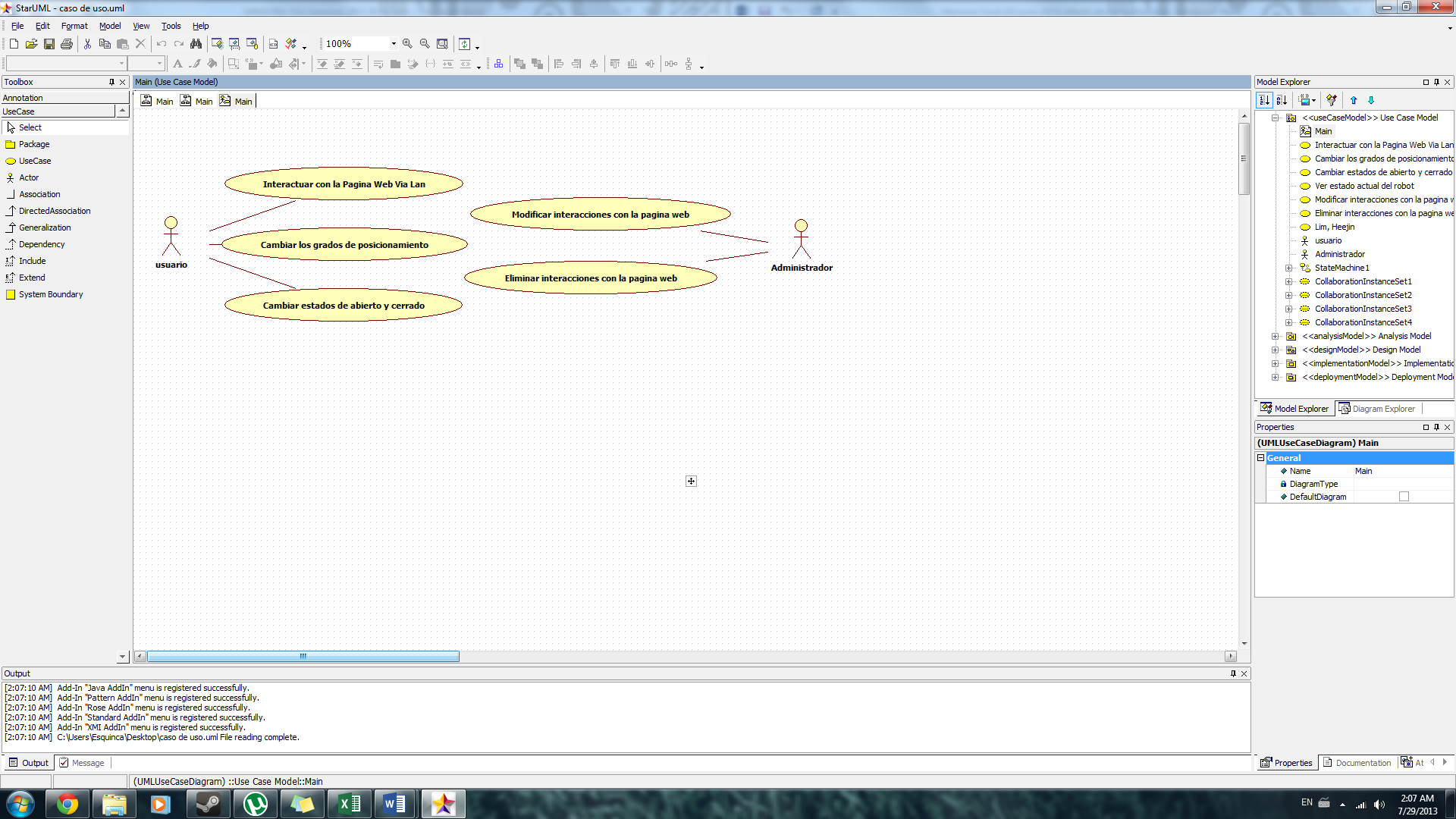
En la ilustración 12, se puede observar los actores que participan en la interacción con el robot. Por un lado se tiene al usuario que es el único que interactuara directamente con el robot, lo podrá realizar vía LAN a través de una interfaz web hosteada por el micro controlador. Por el otro lado el administrador solo estará en contacto con el micro controlador, ya que el tendrá el acceso para cambiar los estados de interacción ya que si alguno de estos falla o está en mantenimiento los podrá desactivar para que el que si sirva esté funcionando en ese momento. También, en caso que la empresa lo requiera, podrá agregar o eliminar las interacciones o módulos que en ese momento se encuentren programados en el robot.

Ilustración . Casos de uso

|  |  |
| --- | --- |
| Nombre | Interacción con la página Web vía LAN |
| Actor | Usuario |
| Descripción | Se refiere a la interacción del usuario con el robot mediante la interfaz web. |
| Disparador | Ingresar a la interfaz web |
| Precondición | Ninguna |
| Postcondición | Ninguna |
| Flujo Normal | 1-El usuario ingresa a la página web.  2-El usuario ingresa datos.  3-Validación de datos.  4-Microcontrolador Responde.  5-Termina caso de uso. |
| Flujo Alternativo | 1-El microcontrolador manda un mensaje de error.  2-La interfaz manda el mensaje.  3-Termina caso de uso. |
| Excepciones | Ninguna |
| Prioridad | Alta |
| Frecuencia de uso | Alta |

Los siguientes cuadros de descripción (ilustración 14, 15, 16, 17) que se muestran a continuación, corresponden a los 5 casos de uso mostrados en la ilustración 14.

Caso de uso “Interacción vía LAN”

Ilustración . Cuadro de descripción vía WEB

Caso de uso “Cambio de grados”

|  |  |
| --- | --- |
| Nombre | Cambio de grados |
| Actor | Usuario |
| Descripción | Se refiere a la interacción del usuario con el robot mediante la interfaz web el cual ingresará datos para el microcontrolador los procese. |
| Disparador | Mandar los datos (grados) |
| Precondición | Ninguna |
| Postcondición | Ninguna |
| Flujo Normal | 1-El usuario ingresa a la página web.  2-El usuario ingresa datos.  3-Validación de datos.  4-Microcontrolador Responde.  5-Termina caso de uso. |
| Flujo Alternativo | 1-El microcontrolador manda un mensaje de error.  2-La interfaz manda el mensaje.  3-Termina caso de uso. |
| Excepciones | Ninguna |
| Prioridad | Alta |
| Frecuencia de uso | Alta |

Ilustración . Cuadro de descripción "Cambio de grados"

Caso de uso “Modificar interacciones”

|  |  |
| --- | --- |
| Nombre | Modificar interacciones. |
| Actor | Administrador |
| Descripción | Se refiere al agregar o modificar nuevos módulos. |
| Disparador | Ninguno |
| Precondición | Ninguna |
| Postcondición | Ninguna |
| Flujo Normal | 1-El administrador modifica el código del servidor (Microcontrolador)  2-El microcontrolador interpreta el nuevo código  3-Microcontrolador Responde.  4-Retroalimenta  5-Termina caso de uso |
| Flujo Alternativo | No aplica |
| Excepciones | Ninguna |
| Prioridad | Alta |
| Frecuencia de uso | Alta |

Ilustración . Cuadro de descripción "Modificar interacciones."

Caso de uso “Eliminar interacciones.”

|  |  |
| --- | --- |
| Nombre | Eliminar interacciones. |
| Actor | Administrador |
| Descripción | Se refiere al Eliminar módulos o código obsoleto. |
| Disparador | Ninguno |
| Precondición | Ninguna |
| Postcondición | Ninguna |
| Flujo Normal | 1-El administrador Elimina el código del servidor (Microcontrolador)  2-El microcontrolador interpreta el nuevo código  3-Microcontrolador Responde.  4-Retroalimenta  5-Termina caso de uso |
| Flujo Alternativo | No aplica |
| Excepciones | Ninguna |
| Prioridad | Alta |
| Frecuencia de uso | Media |

Ilustración . Cuadro de descripción "Eliminar interacciones"

* 1. **Diagramas de secuencia**

En la ilustración 18, se muestra el diagrama “Interacción vía LAN”, en este se observa la secuencia que realiza el usuario para obtener una respuesta en la interfaz web a través del micro controlador.

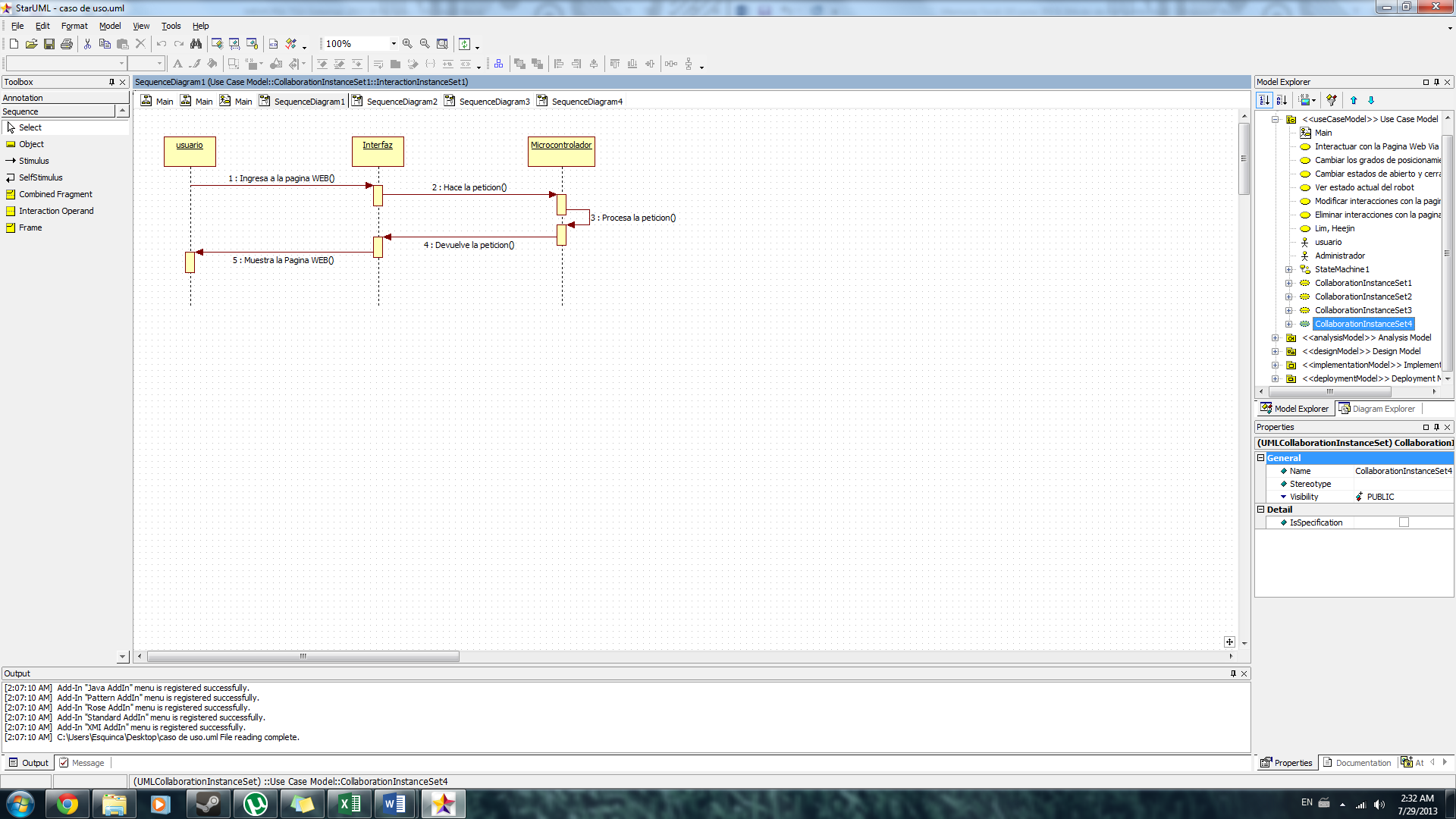


Ilustración . Diagrama de secuencia "Interacción vía WEB"

En la ilustración 19, se muestran los pasos que debe realizar el usuario para modificar los grados del robot a los que se desea moverlo.

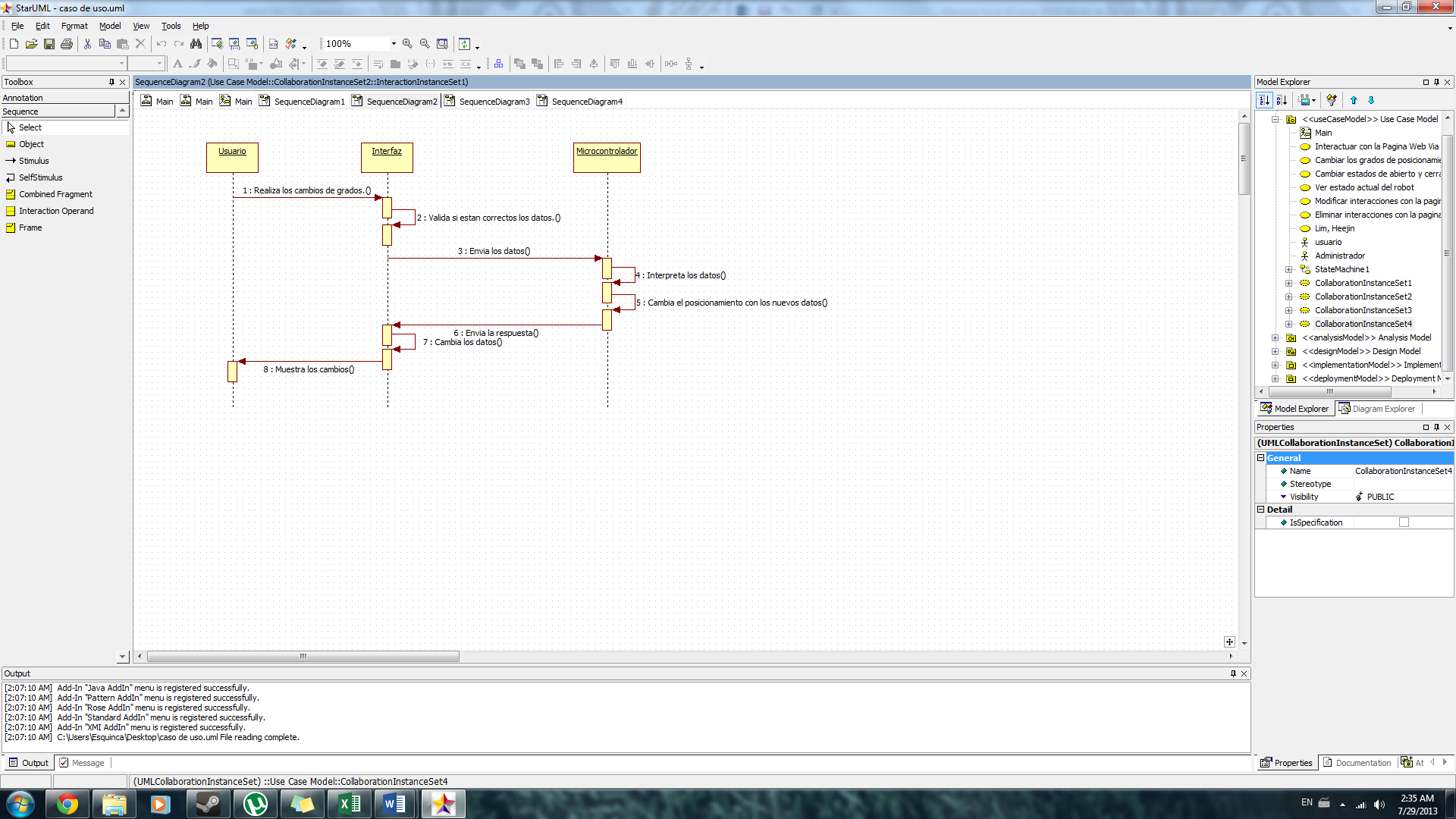


Ilustración . Diagrama de secuencia "Cambio de grados"

El diagrama “Modificar interacciones” (ilustración 20), muestra la secuencia que debe seguir el administrador para agregar o modificar módulos. Esta opción es para que la empresa que lo adquiera pueda solicitar cambios y actualizaciones a la información que quiere que el robot realice.

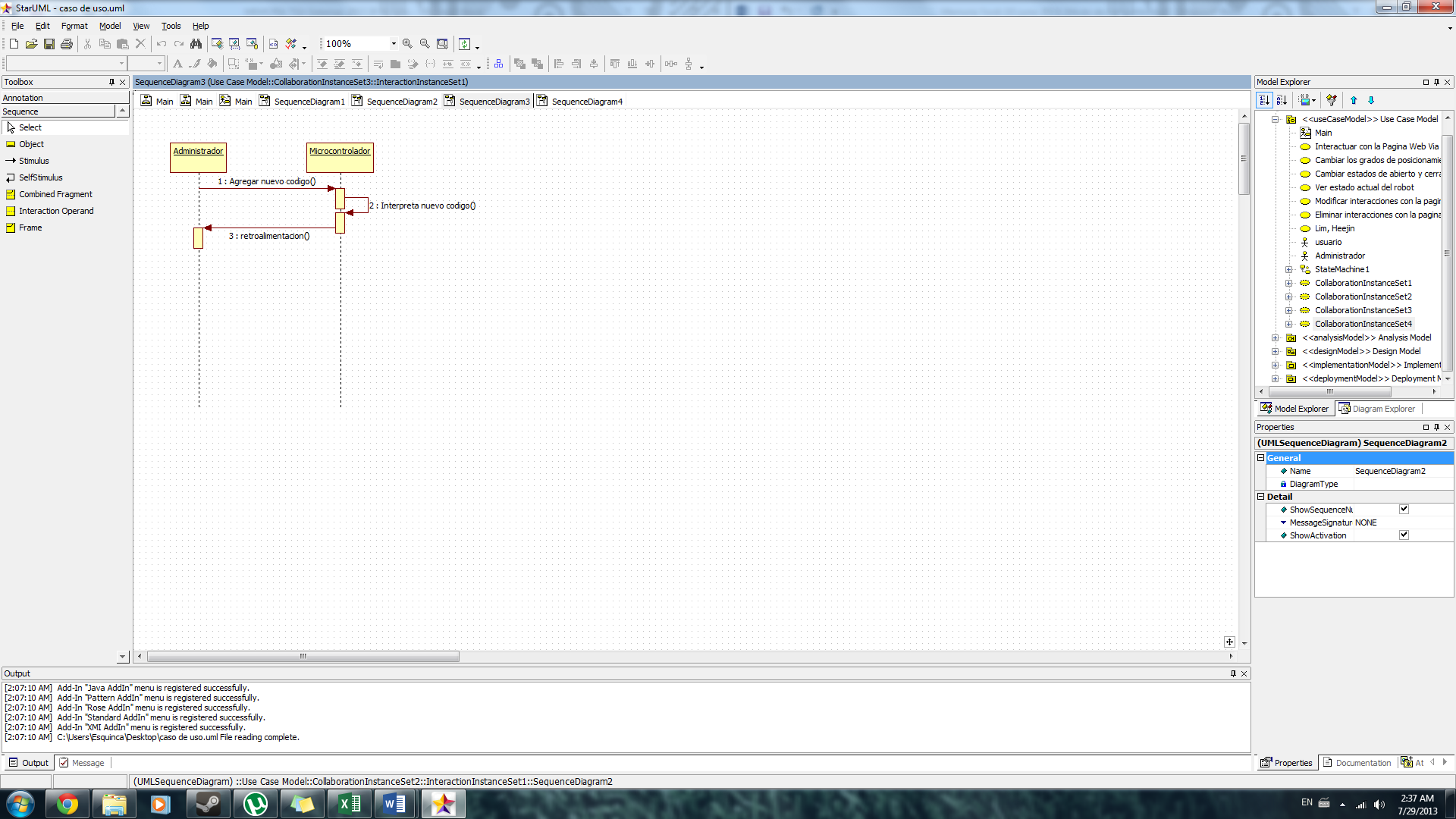


Ilustración . Diagrama de secuencia "Modificar Interacciones"

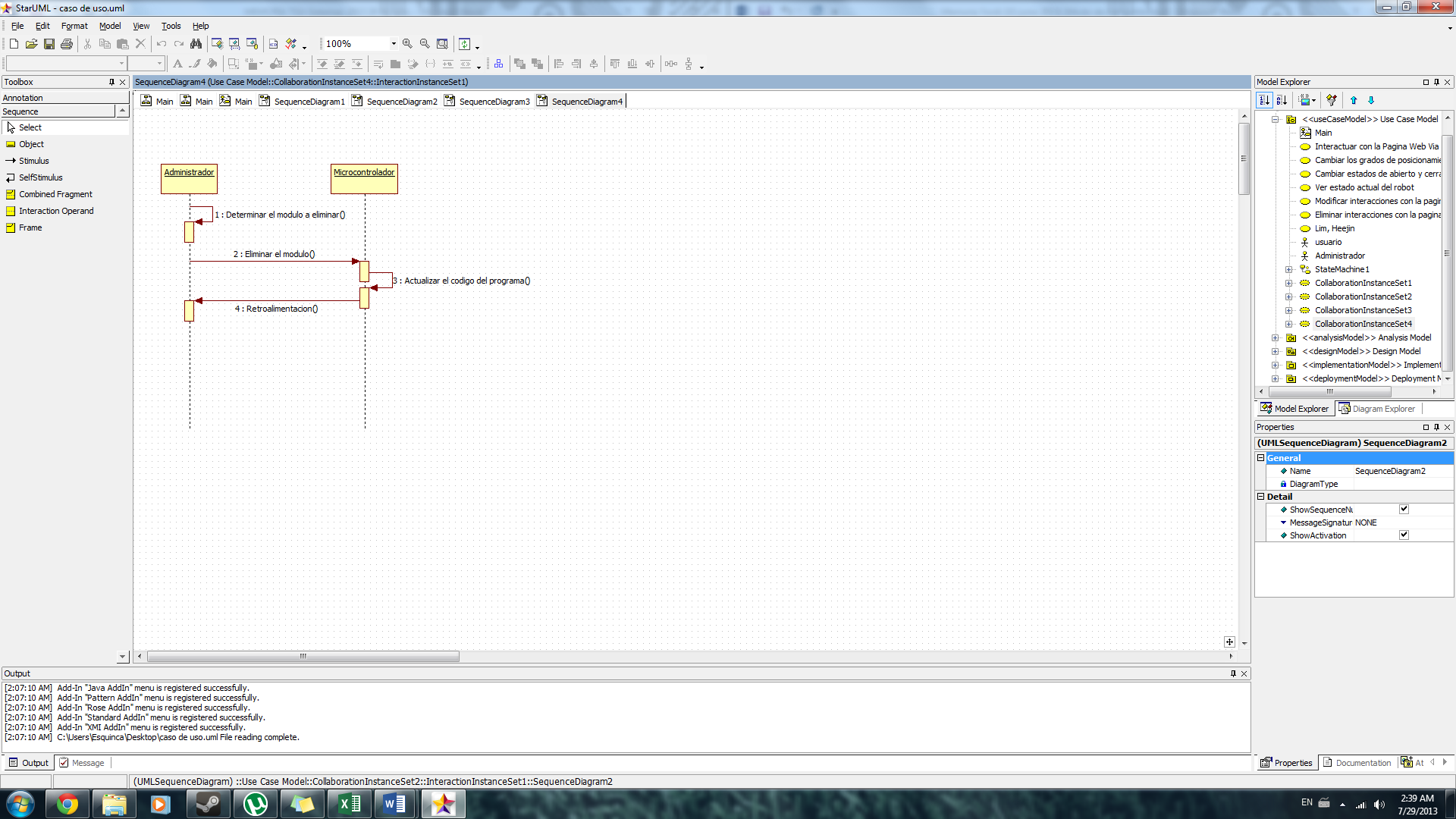
El diagrama “Eliminar Interacciones” (ilustración 21), muestra la secuencia que debe seguir el administrador para eliminar un comando de voz. Esta opción es para que la empresa que lo adquiera pueda solicitar que se eliminen comandos obsoletos o desactualizados

Ilustración . Diagrama de secuencia "Eliminar interacciones"

* 1. **Diagramas de actividades**

El diagrama de actividades “Interacción Vía WEB” (ilustración 22), explica de manera simple el proceso que se lleva acabo al interactuar con el microcontrolador.

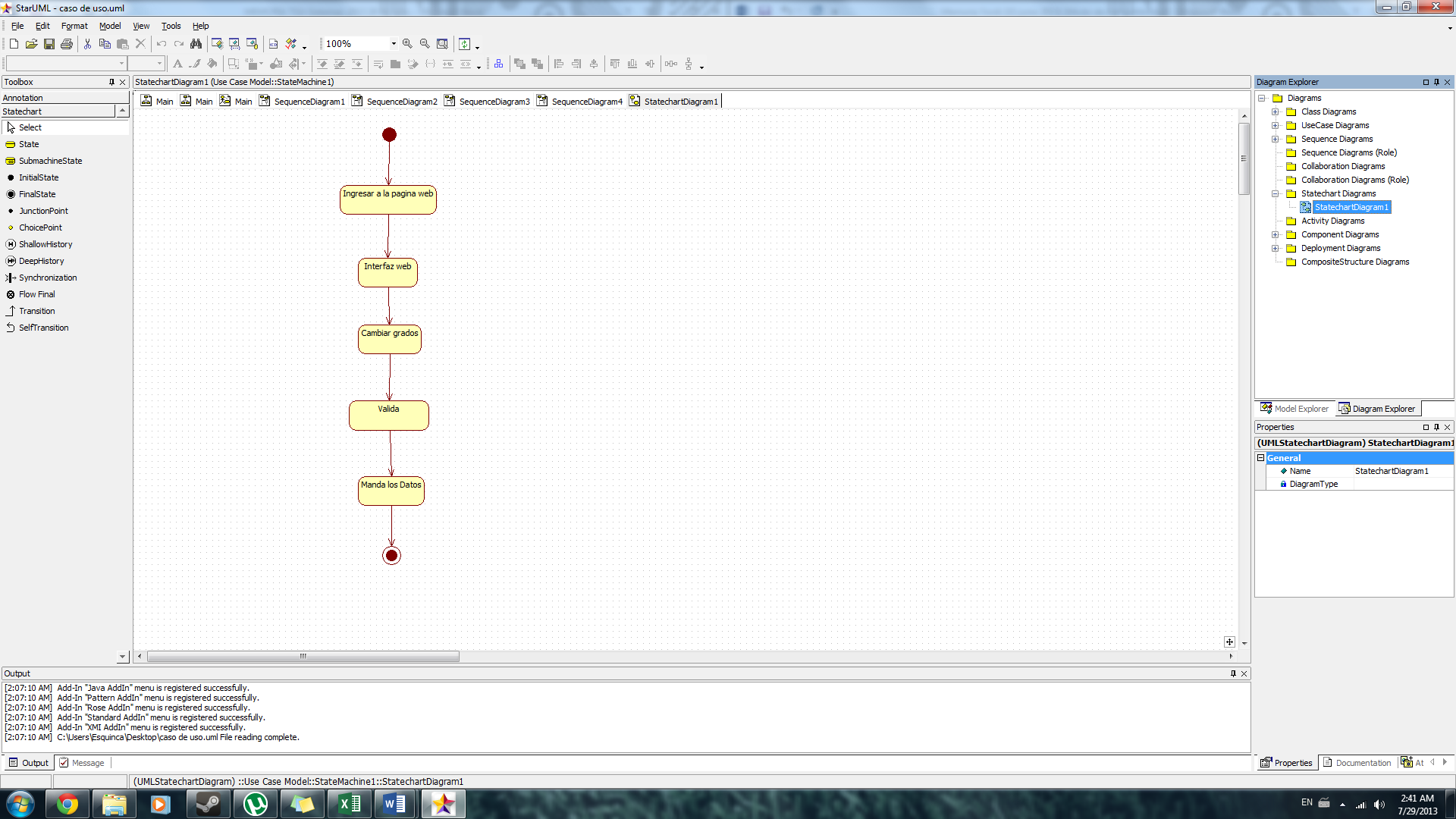


Ilustración Diagrama de actividades

V.3 Herramientas de desarrollo

1. **Lenguaje de programación utilizado**

Para realizar la interfaz web se eligió el lenguaje de programación HTML5 gracias a las características que ofrece en cuanto a manejo de información, validación de información e incluso mostrar y exhibir información. Además de que en tan poco tiempo que fue lanzado se ha convertido en uno de los estándares en el diseño de páginas web modernas se le agregaron muchas funciones nuevas que hace que la programación sea de manera eficaz y simple.

Por otro lado, el lenguaje utilizado para controlar la placa Arduino es basado en lenguaje C, por lo tanto es muy versátil y fácil de programar.

1. **Pruebas de caja negra.**

En la (ilustración 23), explica la forma de realizar de la prueba de ejecución de la página web.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1. INFORMACIÓN DEL MÓDULO DE PRUEBA** | |  | | | | | | |
| NOMBRE DEL MÓDULO DE PRUEBA | Ejecutar la página web | | FECHA DE EJECUCIÓN | | 7/15/2013 | # DEL MÓDULO DE PRUEBA | | 1 |
|
| DESCRIPCIÓN | | Conectar y establecer la IP y la MAC del servidor en el Arduino y escribir en el navegador la IP previamente establecida. | | | | | | |
|
| **2. MÓDULO DE PRUEBA** | | |  | | | | | |
| 2.1 PRECONDICIONES |  | | 2.2 PASOS DE LA PRUEBA | | |  | | |
| 1. Tener el Ethernet shield puesto. | | | 1. Conectar el Arduino. | | | | | |
| 2. Tener prendido el Router. | | | 2. Establecer la IP y la MAC del servidor. | | | | | |
| 3. Tener almacenada la página web en la memoria SD. | | | 3. Conectar la memoria SD al shield. | | | | | |
|  | | | 4. Correr el código. | | | | | |
|  | | | 5. Ingresar la IP en el navegador. | | | | | |
|  | | |  | | | | | |
|  | | |  | | | | | |
| **3. RESULTADOS DE LA PRUEBA** | |  |  |  |  |  |  |  |
| 3.1 RESULTADO ESPERADO | Ejecución de la pagina | | 3.2 RESULTADO OBTENIDO | | Se ejecutó la pagina | | 3.3 VEREDICTO | Exitosa |
|
| **4. REVISIÓN** | |  | | | | | | |
| 4.1 REALIZADOR DE LA PRUEBA | Jose Antonio Esquinca Bonilla | | 4.2 OBSERVACIONES | | Ninguna. | | | |
|
|

Ilustración . Prueba de caja Negra #1

En la ilustración 24 se muestra la prueba de caja de negra #1 donde se ejecuta la página web alojada en la Memoria SD del Ethernet shield conectado al Arduino.

Ilustración . Muestra de la prueba #1

En la ilustración 25 se muestra la misma prueba (ilustración 24), solo que en este caso el resultado no fue el mismo.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1. INFORMACIÓN DEL MÓDULO DE PRUEBA** | |  | | | | | | |
| NOMBRE DEL MÓDULO DE PRUEBA | Ejecutar la página web | | FECHA DE EJECUCIÓN | | 7/22/2013 | # DEL MÓDULO DE PRUEBA | | 2 |
|
| DESCRIPCIÓN | Conectar y establecer la IP y la MAC del servidor en el Arduino y escribir en el navegador la IP previamente establecida. | | | | | | | |
|
| **2. MÓDULO DE PRUEBA** | |  | | | | | | |
| 2.1 PRECONDICIONES |  | | 2.2 PASOS DE LA PRUEBA | | |  | | |
| 1. Tener el Ethernet shield puesto. | | | 1. Conectar el Arduino. | | | | | |
| 2. Tener prendido el Router. | | | 2. Establecer la IP y la MAC del servidor. | | | | | |
| 3. Tener almacenada la página web en la memoria SD. | | | 3. Conectar la memoria SD al shield. | | | | | |
|  | | | 4. Correr el código. | | | | | |
|  | | | 5. Ingresar la IP en el navegador. | | | | | |
|  | | |  | | | | | |
|  | | |  | | | | | |
| **3. RESULTADOS DE LA PRUEBA** | |  |  |  |  |  |  |  |
| 3.1 RESULTADO ESPERADO | Ejecución de la pagina | | 3.2 RESULTADO OBTENIDO | | No se mostró la página. | | 3.3 VEREDICTO | Fracaso |
|
| **4. REVISIÓN** | |  | | | | | | |
| 4.1 REALIZADOR DE LA PRUEBA | Jose Antonio Esquinca Bonilla | | 4.2 OBSERVACIONES | | Modificar Código HTML | | | |
|
|

Ilustración . Prueba de caja Negra #2

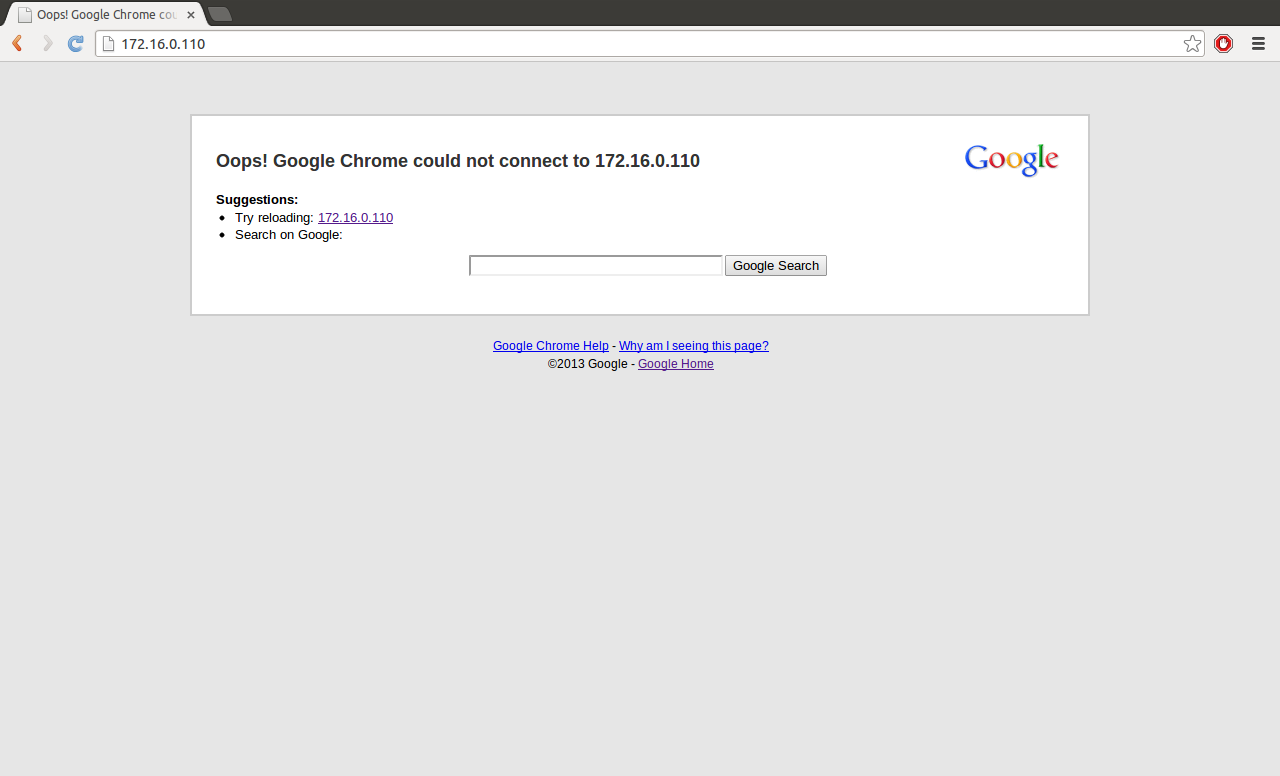
Al tratar de entrar a la página web el explorador arrojó un error el cual se puede ver en la ilustración 26. La razón fue que el microcontrolador no estaba detectando el archivo HTML.

Ilustración . Muestra de la prueba #2

La prueba (ilustración 27) se realizó para probar la ejecución de la cadena PHP que se manda después de darle clic al botón play en la página Web.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1. INFORMACIÓN DEL MÓDULO DE PRUEBA** | |  | | | | | | |
| NOMBRE DEL MÓDULO DE PRUEBA | Ejecución de la cadena | | FECHA DE EJECUCIÓN | | 7/22/2013 | # DEL MÓDULO DE PRUEBA | | 3 |
|
| DESCRIPCIÓN | Ingresar los grados a los que se quiera mover al robot en la página WEB. | | | | | | | |
|
| **2. MÓDULO DE PRUEBA** | |  | | | | | | |
| 2.1 PRECONDICIONES |  | | 2.2 PASOS DE LA PRUEBA | | |  | | |
| 1. Tener el Ethernet shield puesto. | | | 1. Conectar el Arduino. | | | | | |
| 2. Tener prendido el Router. | | | 2. Conectar la memoria SD al shield. | | | | | |
| 3. Tener almacenada la página web en la memoria SD. | | | 3. Ingresar a la página web con la IP previamente establecida. | | | | | |
|  | | | 4. Ingresar los grados. | | | | | |
|  | | | 5. Dar clic a "Play" | | | | | |
|  | | |  | | | | | |
|  | | |  | | | | | |
| **3. RESULTADOS DE LA PRUEBA** | |  |  |  |  |  |  |  |
| 3.1 RESULTADO ESPERADO | Retorno de la cadena PHP | | 3.2 RESULTADO OBTENIDO | | Devolvió la cadena | | 3.3 VEREDICTO | Exitosa |
|
| **4. REVISIÓN** | |  | | | | | | |
| 4.1 REALIZADOR DE LA PRUEBA | Jose Antonio Esquinca Bonilla | | 4.2 OBSERVACIONES | | | Ninguna. | | |
|
|

Ilustración . Prueba de caja Negra #3

Como se puede apreciar en la ilustración 28, en la URL de la página se encuentra una cadena correctamente producida por el código php, la cual es usada por el microcontrolador para extraer datos.

Ilustración . Muestra de la prueba #3

La prueba (ilustración 29) se realizó para probar el comportamiento y extracción de la cadena en el microcontrolador.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1. INFORMACIÓN DEL MÓDULO DE PRUEBA** | |  | | | | | | |
| NOMBRE DEL MÓDULO DE PRUEBA | Análisis de la cadena | | FECHA DE EJECUCIÓN | | 7/22/2013 | # DEL MÓDULO DE PRUEBA | | 4 |
|
| DESCRIPCIÓN | El Arduino debe recibir la cadena y descomponerla para recibir los valores para así poder manipularlos. | | | | | | | |
|
| **2. MÓDULO DE PRUEBA** | |  | | | | | | |
| 2.1 PRECONDICIONES |  | | 2.2 PASOS DE LA PRUEBA | | |  | | |
| 1. Tener el Ethernet shield puesto. | | | 1. Conectar el Arduino. | | | | | |
| 2. Tener prendido el Router. | | | 2. Conectar la memoria SD al shield. | | | | | |
| 3. Tener almacenada la página web en la memoria SD. | | | 3. Mandar el request con los datos. | | | | | |
| 4. Ingresar a la página WEB | | |  | | | | | |
|  | | |  | | | | | |
|  | | |  | | | | | |
|  | | |  | | | | | |
| **3. RESULTADOS DE LA PRUEBA** | |  |  |  |  |  |  |  |
| 3.1 RESULTADO ESPERADO | Extracción de valores | | 3.2 RESULTADO OBTENIDO | | Extrajo los valores | | 3.3 VEREDICTO | Exitosa |
|
| **4. REVISIÓN** | |  | | | | | | |
| 4.1 REALIZADOR DE LA PRUEBA | Jose Antonio Esquinca Bonilla | | 4.2 OBSERVACIONES | | Ninguna. | | | |
|
|

Ilustración . Prueba de caja Negra #4

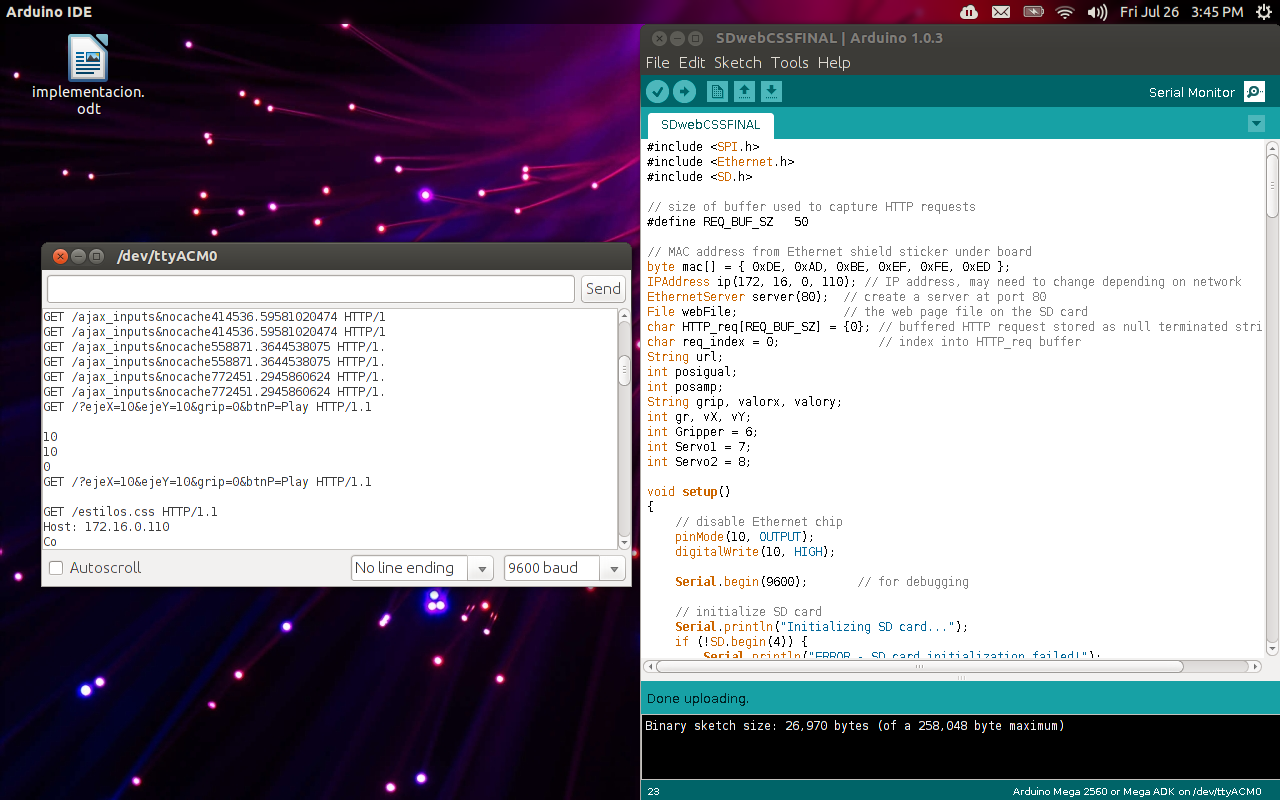
Como ya había explicado anteriormente, aquí se muestra el Arduino recibiendo dicha cadena y extrayendo los valores de la página para poder procesarlos y asignarlos al robot.

Ilustración . Muestra de la prueba #4

V.4 Implementación del proyecto

1. **Instalación del sistema / Respaldo de información**

Como ya se había mencionado con anterioridad, el micro controlador será el que realice todo el trabajo pesado ya que se encargará de mantener la interfaz web activada o desactivada. Todo este sistema que sucede detrás, la facilidad de acceso a las funciones necesarias y comandos utilizados se tiene gracias a librerías (ilustración 31) que fueron hechas específicamente para controlar estos dispositivos auxiliares que permiten realizar las características principales de este proyecto.

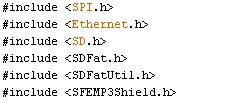
****

Ilustración . Librerías

Una de las dos partes esenciales en un sketch o programa de Arduino es el “setup” ya que éste prepara todas las variables, actuadores, sensores, shields, etc. Tal como se puede observar en la ilustración 32, en la cual muestra dos cosas muy importantes la revisión e inicialización de la memoria SD que es la que se encarga de tener almacenadas la página web con el CSS, las imágenes y por el otro lado la activación del servidor web ingresando la variable que contendrá la IP y la MAC del micro controlador.



Ilustración . void setup ()

Después sigue el “loop” que es el que contiene el código del programa que se estará repitiendo. En esta primera parte (ilustración 33), se observa cómo se inicializan las peticiones solo si el cliente está activo, ósea si ingresa a la interfaz web desde un navegador. Enseguida dos funciones muy importantes que se revisaran más adelante, WaitForRequest (ilustración 35) y ParseReceivedRequest (ilustración 36).

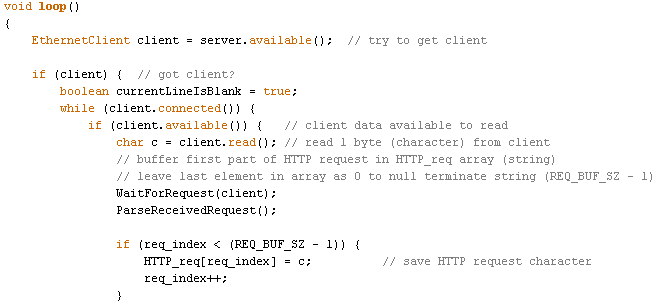


Ilustración . void loop () parte 1

En la ilustración 34 se puede identificar la segunda parte del código. La cual contiene otra función muy importante de la cual se hablará adelante también que es XML response (ilustración 37). Después se observa como busca en la memoria SD la página web que en este caso es “index.htm” y la abre.



Ilustración . void loop () parte 2

WaitForRequest (ilustración 35) es la primer función que se observa en el “loop” ya que esta es la que espera que dentro de la interfaz web haya sucedido una petición o en este caso que el usuario haya seleccionado la pregunta que desea realizar. Esta almacenará en el array Buffer la cadena completa en donde vendrán los valores que necesitamos para poder decidir que accion tiene que hacer el robot. El único parámetro que pide es el cliente que esta conectado o en términos del proyecto la interfaz Web.

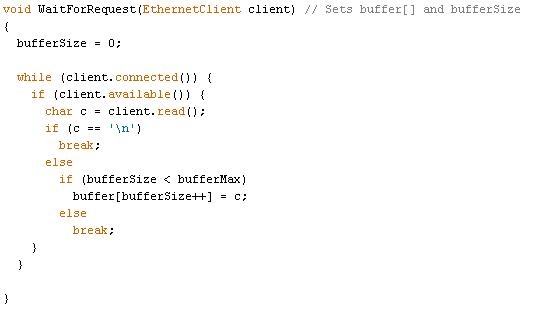


Ilustración . WaitForRequest

Una vez que se tiene almacenada la petición que la interfaz web envio, la siguiente función que entra en juego es ParseReceivedRequest (ilustración 36).

Esta funcion no hará mas que cortar la cadena recibida o separar la información que utilizará para reproducir la acción del robot. La petición recibida tendrá el simbolo de “&” para separar cada valor, por lo tanto con funciones de String se podrá acceder a ellos.

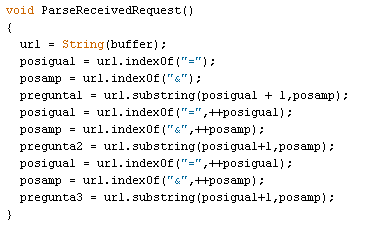


Ilustración . ParseReceivedRequest

La última función que tiene el codigo del programa es la de XMLResponse (ilustración 37), que esta lo que realiza es actualizar los valores en tiempo real en la interfaz web del estado del Robot.

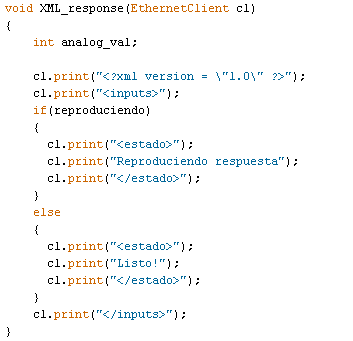


Ilustración . XML\_response

V.5 Análisis de resultados.

Para comprobar el funcionamiento óptimo del robot se probaron todos los mecanismos y funciones disponibles como se puede observar en la ilustración 38, en la cual se muestra la página web cargada al 100% desde el micro controlador con todas las funciones habilitadas (CSS, imágenes) en red de área local y extrayendo los datos los datos del robot de su posición actual.

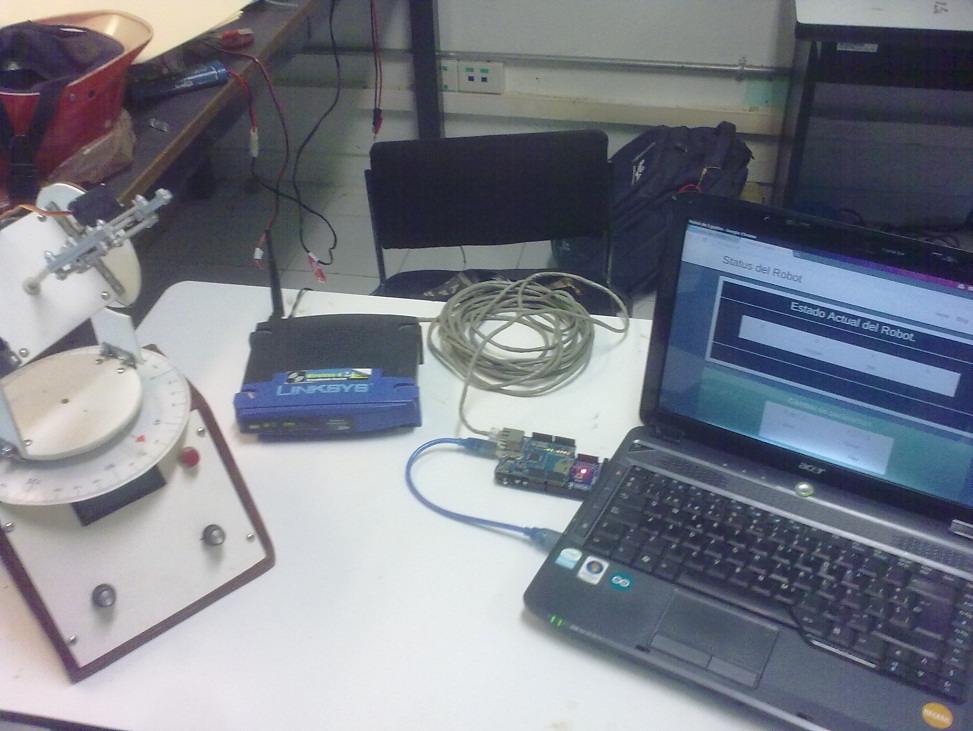


Ilustración . Proyecto funcionando.

Para la interfaz web, se utilizó la herramienta CSS para aprovechar al máximo las funciones del micro controlador quepermite, por medio de una memoria SD guardar cualquier tipo de archivos. Esto permitió agregar archivos CSS para darle apariencia moderna y simple (ilustración 39).



Ilustración . Interfaz Amigable



El micro controlador se utilizó de manera eficaz y cumplió perfectamente con el propósito de actuar como servidor de la interfaz la cual carga rápidamente los elementos de la interfaz y funciona de manera correcta. Como se puede observar en la ilustración 40 el micro controlador tiene conectado un cable Ethernet el cual proviene de un Router y se le asigna un IP para poder acceder al mismo.

Ilustración . Servidor

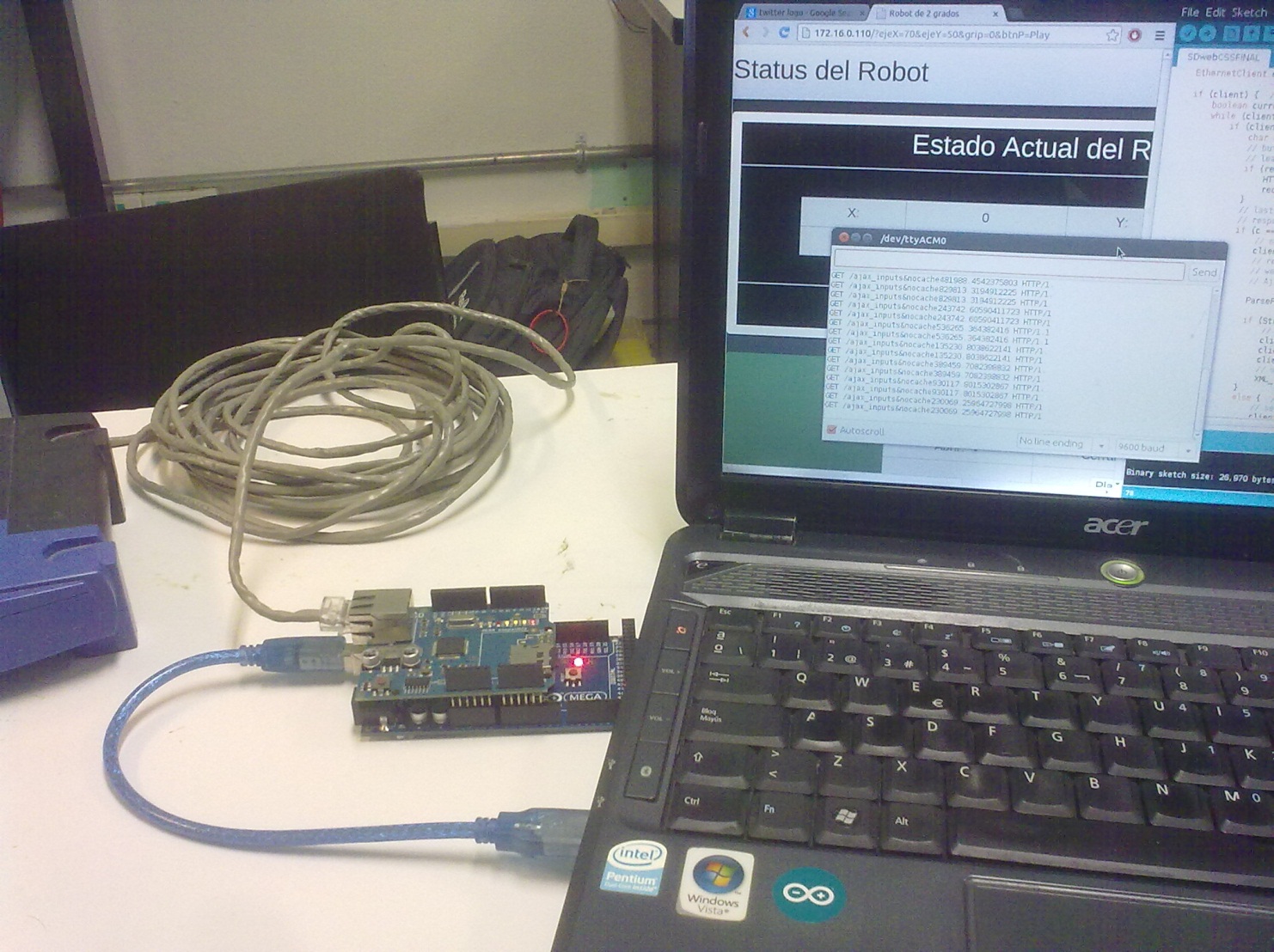
La programación del micro controlador aporta estabilidad, rapidez y estabilidad en el proyecto ya realiza operaciones muy completas. Recibe y envía cadenas de la página WEB y las interpreta para poder mandar esos datos al robot de forma simple y rápida, en la ilustración 41 se presenta un ejemplo.

Ilustración . Programación del microcontrolador

V.6 Estudio de factibilidad

1. Sustento Operativo

El proyecto está enfocado a todo tipo de usuario, la interfaz Web es amigable e intuitiva por lo que no se requieren conocimientos previos para utilizarla, además se incluirá un manual de usuario de la interfaz.

1. Sustento Técnico

La universidad cuenta con la infraestructura y herramientas para el desarrollo del proyecto, una vez concluido el usuario solo requerirá una computadora con acceso a internet.

1. Sustento Económico

Para el desarrollo de este proyecto no fue necesaria ninguna inversión por parte de la universidad ya que se contaba con las herramientas que se mencionaron en el apartado de fundamentos de esta Memoria.

1. Sustento Social

La aportación del proyecto es la posibilidad de manipular remotamente cualquier dispositivo que pueda soportar el Arduino, en este caso un robot de 2 grados de libertad, ofreciendo al usuario la facilidad de operar dispositivos desde cualquier ubicación.

1. CONCLUSIONES Y CONTRIBUCIÓN TÉCNICA

**Conclusiones**

Es común que en el desarrollo de proyectos existan dificultades en las últimas etapas del mismo. La manera de evitar que esto pase es delimitar siempre el proyecto que se desarrollará, realizar un buen análisis desde el principio, que los objetivos sean congruentes, medibles y alcanzables, administrar y distribuir bien el tiempo en el cronograma de actividades siempre dejando un pequeño espacio para imprevistos. Para que al final te sea posible terminar en tiempo y forma.

Basándose en esto se tienen las siguientes conclusiones:

* Arduino a pesar de no tener un procesador grande, no tiene problemas en hostear una página Web con CSS.
* La investigación y análisis realizado para llegar al primer prototipo ha dejado una base de conocimientos amplia para poder desarrollar otras ideas.
* Para programar un sistema embebido Arduino una vez más demuestra que tiene la capacidad de realizar varias tareas al mismo tiempo sin tener fallas a largo plazo y en caso de tener algún error o si se requiere realizar un cambio de parámetros tiene fácil acceso y le facilita el trabajo al desarrollador.
* Es bueno tener conocimientos en lenguaje C, ya que por ser fácil de entender, versátil y extenso las nuevas tecnologías y las interfaces para programar sus componentes se basan en él.

**Contribuciones**

A lo largo del desarrollo del proyecto se ha podido establecer que este mismo tendrá un fuerte impacto en distintos sectores de los que se ha redactado a lo largo del documento.

Desarrolladores

* Adquisición de nuevos conocimientos
* Creación de áreas de oportunidad.

Turístico

* Creación de nuevos servicios para sus clientes.
* Pauta para nuevas ideas de proyectos a futuro.
* Impacto directo en la economía.

Tecnológico

* Creación de nuevas fuentes de investigación.
* La unión de conocimientos de las diferentes ramas de la tecnología.
* Innovación en programación embebida.
* Atracción de recursos para desarrollo de proyectos.

BIBLIOGRAFÍA Y CONSULTAS REALIZADAS

**Arduino. 2005.** Arduino. [En línea] 21 de 04 de 2005. http://arduino.cc/en/.

**Barr, Michael. 2007.** *Embedded systems design.* 2007.

**Hangar. 2007.** hangar.org. [En línea] 28 de 04 de 2007. www.hangar.org.

**Kevan, Tom. 2013.** Sensors. [En línea] 15 de 07 de 2013. http://www.sensorsmag.com.

**Kosow, Irving L. 2011.** *Máquinas eléctricas y transformadores.* 2011.

**SparkFun. 2008.** SparkFun. [En línea] 10 de 05 de 2008. https://www.sparkfun.com.

ANEXOS

Anexo 1. TABLA DE FIGURAS O ILUSTRACIONES

[Ilustración 1 Organigrama de la Universidad 14](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240498)

[Ilustración 2. Infraestructura 15](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240499)

[Ilustración 3. Sistema embebido 21](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240500)

[Ilustración 4. Placa Arduino 22](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240501)

[Ilustración 5. Placa Arduino Uno 23](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240502)

[Ilustración 6. Ethernet Shield 24](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240503)

[Ilustración 7. Motor 25](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240504)

[Ilustración 8. Actuador Neumático 25](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240505)

[Ilustración 9. ServoMotor 26](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240506)

[Ilustración 10. Sensores 27](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240507)

[Ilustración 11. Dispositivo NC 28](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240508)

[Ilustración 12. Etapas del modelo de prototipos 30](#_Toc363240509)

[Ilustración 13. Casos de uso 32](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240510)

[Ilustración 14. Cuadro de descripción vía WEB 33](#_Toc363240511)

[Ilustración 15. Cuadro de descripción "Cambio de grados" 34](#_Toc363240512)

[Ilustración 16. Cuadro de descripción "Modificar interacciones." 34](#_Toc363240513)

[Ilustración 17. Cuadro de descripción "Eliminar interacciones" 34](#_Toc363240514)

[Ilustración 18. Diagrama de secuencia "Interacción vía WEB" 35](#_Toc363240515)

[Ilustración 19. Diagrama de secuencia "Cambio de grados" 35](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240516)

[Ilustración 20. Diagrama de secuencia "Modificar Interacciones" 36](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240517)

[Ilustración 21. Diagrama de secuencia "Eliminar interacciones" 36](#_Toc363240518)

[Ilustración 22 Diagrama de actividades 37](#_Toc363240519)

[Ilustración 23. Prueba de caja Negra #1 39](#_Toc363240520)

[Ilustración 24. Muestra de la prueba #1 39](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240521)

[Ilustración 25. Prueba de caja Negra #2 40](#_Toc363240522)

[Ilustración 26. Muestra de la prueba #2 40](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240523)

[Ilustración 27. Prueba de caja Negra #3 41](#_Toc363240524)

[Ilustración 28. Muestra de la prueba #3 41](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240525)

[Ilustración 29. Prueba de caja Negra #4 42](#_Toc363240526)

[Ilustración 30. Muestra de la prueba #4 42](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240527)

[Ilustración 31. Librerías 43](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240528)

[Ilustración 32. void setup () 44](#_Toc363240529)

[Ilustración 33. void loop () parte 1 45](#_Toc363240530)

[Ilustración 34. void loop () parte 2 46](#_Toc363240531)

[Ilustración 35. WaitForRequest 47](#_Toc363240532)

[Ilustración 36. ParseReceivedRequest 48](#_Toc363240533)

[Ilustración 37. XML\_response 49](#_Toc363240534)

[Ilustración 38. Proyecto funcionando. 50](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240535)

[Ilustración 39. Interfaz Amigable 51](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240536)

[Ilustración 40. Servidor 51](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240537)

[Ilustración 41. Programación del microcontrolador 52](file:///C:\Users\Eternal\Desktop\MEMORIA%20TSU-Sistemas%20%202013%2029%20de%20julio..docx#_Toc363240538)

|  |  |
| --- | --- |
| Arduino | Arduino es una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles y fáciles de usar. (Arduino, 2005) |
| C++ | Es un lenguaje de programación diseñado a mediados de los años 1980. Con mecanismos que permitan la manipulación de objetos. (Stroustrup, 1997) |
| CSS | Son hojas de estilo en cascada o (Cascading Style Sheets, o sus siglas CSS) hacen referencia a un lenguaje de hojas de estilos usado para describir la presentación semántica (el aspecto y formato) de un documento escrito en lenguaje de marcas. (W3C, 2009) |
| HTML | Lenguaje de marcado hipertextual, hace referencia al lenguaje de marcado predominante para la elaboración de páginas web que se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes. (Mora, 2001) |
| IP | Una dirección IP es una etiqueta numérica que identifica, de manera lógica y jerárquica, a una interfaz de red. |
| MAC | Es un identificador de 48 bits (6 bloques hexadecimales) que corresponde de forma única a una tarjeta o dispositivo de red. |
| PHP | Es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. |
| Sistema Embebido | Es un sistema de computación diseñado para realizar una o algunas pocas funciones dedicadasfrecuentemente en un sistema. de computación en tiempo real. |

GLOSARIO