

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA

MAESTRÍA Y DOCTORADO EN CIENCIAS E INGENIERÍA



“USO DE PHIDGETS EN LA ENSEÑANZA DE CONCEPTOS
INTRODUCTORIOS EN LAS DISCIPLINAS DE COMPUTACIÓN Y
ELECTRÓNICA”

TESIS

Que para obtener el grado de:
MAESTRO EN CIENCIAS
Presenta:

DANIEL SALVADOR VIDAL SÁNCHEZ

DIRECTOR DE TESIS:
M.C. JORGE EDUARDO IBARRA ESQUER

Resumen

En este documento se presenta la posibilidad de utilizar Phidgets para agregar capacidades de interacción que vayan más allá de lo que nos permiten las interfaces de entrada y salida típicas de la computadora. El objetivo de utilizar dispositivos electrónicos, en un ambiente de interacción con la computadora, es ofrecer al usuario mecanismos adicionales para el control de las aplicaciones y para la visualización de los resultados de sus acciones. De esta manera se propone apoyar al proceso de enseñanza-aprendizaje de conceptos introductorios en las disciplinas de computación y electrónica.

Abstract

This document presents the possibility of using Phidgets to add interaction capabilities that go beyond what we are allowed by input and output interfaces typical of the computer. The goal of using electronic devices, in an environment of interaction with the computer, is to offer users additional mechanisms to control applications and to display the results of their actions. This will give support to the teaching-learning process of introductory concepts in computing and electronics disciplines.

Agradecimiento

Quiero agradecer primeramente a Dios que me da las fuerzas para seguir adelante todos los días.

Agradezco a mi familia por su apoyo y comprensión durante este tiempo de estudios de maestría.

Tengo un profundo agradecimiento hacia mis maestros y sinodales, Jorge Ibarra, Brenda Flores, Gabriel López, Rafael Angulo, Cecilia Curlango, por su paciencia, consejos, confianza y presión que me brindaron para poder llegar a hacer todo esto posible. Sobre todo por su tiempo dedicado en la realización de este trabajo.

A mis amigos y compañeros de maestría y doctorado, con los que compartimos momentos difíciles, así como satisfactorios. Por compartir sus recursos, tanto en tiempo como en capacidades.

Finalmente, agradezco al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el apoyo económico brindado durante este tiempo. Agradezco, también por su apoyo, a la Coordinación de Posgrado e Investigación de la UABC, donde este proyecto está registrado con los números de programa 2024 y 105-100.

Dedicatoria

A mi familia y amigos.

Índice General

Capítulo 1. Introducción

1.1 Planteamiento del problema	2
1.2 Antecedentes	2
1.2.1 Déficit de ingenieros	3
1.2.2 Estrategias para combatir el déficit de ingenieros	3
1.2.3 Perfiles educativos de las carreras de ingeniería	4
1.2.4 Sistemas empotrados	4
1.2.5 Aplicaciones interactivas	6
1.3 Objetivos	7
1.3.1 Objetivo general	7
1.3.2 Objetivos específicos	7
1.4 Metodología	8
1.5 Esquema general de la Tesis	8

Capítulo 2. Phidgets

2.1 Definición de Phidgets	9
2.2 Características de Phidgets	9
2.3 Hardware	10
2.3.1 PhidgetInterfaceKit	10
2.3.2 Sensores analógicos	11
2.3.3 PhidgetRFID	13
2.3.4 PhidgetSpatial	14
2.3.5 PhidgetAdvancedServo	15
2.4 Software	16

Capítulo 3. Desarrollo de aplicaciones con Phidgets

3.1 Adaptación de una aplicación para usarse con Phidgets	17
3.1.1 Comprender y especificar el contexto de uso	19
3.1.2 Especificar requerimientos de usuario	19
3.1.3 Producir soluciones de diseño	20
3.1.3.1 Adaptación de la interfaz de entrada	21
3.1.3.2 Descripción del sistema interactivo Space Invaders	23
3.1.3.3 Adaptación del código	26
3.1.4 Evaluación de diseño	28
3.2 Diseño de una aplicación basada en Phidgets	30
3.3 Uso de la biblioteca Phidget	35

Capítulo 4. Evaluación del uso de Phidgets

4.1 Elaboración de herramienta para la evaluación del uso de Phidgets	39
4.2 Evaluación a estudiantes	40
4.3 Análisis e interpretación de resultados	42

Capítulo 5. Propuesta de uso

5.1 Uso de Phidgets para representar conceptos abstractos de computación	44
5.2 Uso de Phidgets para representar conceptos de electrónica	46

Referencias	49
-------------------	----

Capítulo 1

Introducción

1.1 Planteamiento del problema

En estos días a nivel mundial existe un gran déficit de estudiantes ingresando o estudiando en carreras relacionadas a las áreas de la ciencia y la tecnología debido a que para algunos jóvenes les parecen carreras aburridas, teóricas y difíciles [UNESCO, 2004]. Esa tipo de ideas hace que para los jóvenes esas carreras parezcan menos atractivas a la hora de elegir carrera [Marjoram, T., 2010]. En unos países el fenómeno afecta más que en otros, como en la República Mexicana donde el déficit de ingenieros a provocando un rezago en la cantidad de científicos e investigadores que participan activamente en la sociedad [Molina, M., 2011].

Uno de los factores que provoca esta situación es que los jóvenes tienen la percepción errónea de que las carreras técnicas son más difíciles que las demás. Por otro lado, como explica el gerente de Desarrollo de Negocios del Centro de Desarrollo de Software de Intel en Argentina. No hay un claro conocimiento, en los niveles medio superior y superior de educación, acerca de la creciente demanda mundial y local de profesionales [Perazo, C., 2010]. Por otra parte, también se cree que los empleos de ingeniero son mal pagados con respecto al grado de responsabilidad que conllevan [Marjoram, T., 2010].

1.2 Antecedentes

Una de las posibles causas que se le atribuye al fenómeno es la pobre preparación de los estudiantes en ciencias básicas (matemáticas, física, química y biología). Para los estudiantes no es agradable la enseñanza de estas asignaturas, por lo que genera un rechazo a las carreras relacionadas a ellas. Sobre todo, los estudiantes que pasan por la ciencia, enseñada de forma fragmentada, sin lógica, donde lo más importante es la memoria, seguir formulas y no se alcanza una comprensión de porque pasan las cosas, quedan con fobia a la

ciencia [Ulloa, G., 2008].

1.2.1. Déficit de ingenieros

Tanto empresas como instituciones educativas aseguran que es necesario incentivar la formación de una mayor cantidad de ingenieros para cubrir los requerimientos del sector. Las autoridades educativas de Tijuana señalan que se tienen que importar ingenieros ante la falta de talento en la región. De acuerdo al titular de Educación en el Estado de Baja California, en los países desarrollados un promedio de 45% de sus estudiantes ingresan a las ingenierías, en tanto que en México solo del 12 al 15% lo hacen. El déficit es tan grande que otros países exportan ingenieros topógrafos, mineros, petroleros y civiles [El Vigia, 2011].

Paulino Rivas Martínez, director del Tecnológico de Durango, comenta que México está por debajo del requerimiento de ingenieros y que los esfuerzos se deben concentrar en convencer a los jóvenes que están a punto de elegir carrera que opten por las ciencias exactas. De cada 10 jóvenes en México, solo 3.8 se inclinan hacia las ingenierías mientras que en países desarrollados, como Alemania son 8 de cada 10, en Francia casi 7, en Estados Unidos 6. [El Vigia, 2011].

1.2.2. Estrategias para combatir el déficit de ingenieros

Al plantear estrategias para combatir el déficit de ingenieros, uno de los aspectos importantes a considerar es el divulgar a la ciencia y a la tecnología a nivel sociedad. El que los jóvenes se acerquen a la ciencia, el que elijan tomar el papel de desempeñarse profesionalmente en áreas de la producción de alta tecnología, así como la atención de la sociedad presta a los problemas científicos depende en gran medida del grado de popularización de la ciencia. En un estudio realizado por David Berriolope, describe que el uso de la robótica es una de las mejores estrategias en la educación. Los alumnos que han estado involucrados, desde que empiezan sus estudios, con robótica al final escogen una carrera relacionada con ingeniería [Berriolope, D., 2012].

Los museos, festivales, exposiciones, concursos, campamentos temáticos y carteles llamativos dedicados a la ciencia son algunos de los medios de mejorar la adquisición de

conocimientos científicos básicos en una sociedad, en particular entre los niños y jóvenes que vez desprecian más a las carreras científicas [UNESCO].

Por otra parte, se deben de proponer estrategias para evitar la deserción en los estudiantes que ya han optado por una carrera en ingeniería.

1.2.3. Perfiles educativos de las carreras de ingeniería

Tanto la programación como la electrónica básica son consideradas disciplinas diferentes por lo que suelen enseñarse en temas completamente separados. En el sector industrial los procesos para desarrollar dispositivos electrónicos y controles programables son muchas veces combinados en un solo problema de diseño. En estos días muchas personas escriben programas de computadoras de algún tipo. Pero pocos de estos programas se desarrollan para controlar directamente dispositivos electrónicos.

El poder de la programación de microcontroladores, radica en que tiene la capacidad de vincular los programas con dispositivos o circuitos electrónicos con el fin de controlar o analizar la información de ese dispositivo [Johnson, R., 2008].

1.2.4. Sistemas empotrados

Están compuestos por una computadora conectada a dispositivos mecánicos, químicos y eléctricos, son programados para un propósito dedicado y empacados como un sistema completo. Todo sistema eléctrico, mecánico o químico que incluya entradas, decisiones, cálculos, análisis y salidas es un candidato a implementarse como un sistema empotrado [Valvano, J., 2004].

Las funciones que desempeñan estos dispositivos:

- Sensores: Obtienen la información del exterior de acuerdo al fenómeno que se esté observando.
- Interfaces electrónicas: Convierten las señales de un sensor a una forma aceptada por la computadora. De manera similar, convierten las salidas de la computadora a la forma requerida por el sistema que se va a manipular o controlar.
- Software: Efectúa las decisiones, cálculos y análisis necesarios.
- Actuadores: Crean salidas mecánicas o químicas.

Phidgets:

Son la representación física de Widgets. Cuentan con gran variedad de sensores modulares, por ejemplo: temperatura, presión, movimiento, magnetismo, luz, distancia, entre otros.

Los Phidgets son versátiles debido a que su manipulación es de forma directa. Donde a nivel de hardware, solo se conectan al puerto USB. A nivel de software, se puede desarrollar en las plataforma y lenguajes más usados.

WiiMote:

Es el control de mando para la consola Wii. La conexión de este dispositivo es a través de bluetooth por lo cual se puede conectar a cualquier computadora con esta tecnología inalámbrica. Además de los botones cuenta con un sensor óptico infrarrojo y un acelerómetro como sensor de movimiento. Existen varias bibliotecas de programación en lenguajes Java y C/C++.

LEGO Mindstorm:

Kit interactivo de robótica fabricado por LEGO va dirigido a los niños y a aquellos atraídos por el área de la robótica. Se utiliza un bloque NXT donde se almacena el programa que ejecutara el robot, el cual cuenta con una pantalla LCD, botones para manipular el menú, entradas analógicas para leer información de los distintos sensores, y salidas que brindan 9 voltios para manipular actuadores.

Se utiliza el software LEGO Mindstorm NXT 2.0 con un lenguaje de programación basado en bloques llamado NXT-G. Para almacenar los programas se conecta a la computadora mediante la interfaz USB, la versión 2.0 cuenta con tecnología Bluetooth, la cual le permite comunicarse con otras computadoras, sensores, actuadores, otros robots LEGO Mindstorm y otros dispositivos con esta interfaz de comunicación.

Arduino:

Módulos electrónicos con un microprocesador con entradas y salidas digitales, además de entradas analógicas y una salida PWM (modulador de ancho de pulso) de 8 bits para controlar servomotores. El entorno de desarrollo de Arduino esta escrito en Java y corre en Windows, MacOS X y Linux, se puede descargar de su página oficial.

Kinect:

Diseñado para servir como controlador de juegos para la consola Xbox 360. Cuenta con una base giratoria motorizada, una cámara de captación de movimiento, una doble cámara de profundidad 3D, además de cuatro micrófonos que puede reconocer voces por separado.

1.2.5. Aplicaciones interactivas

Los sistemas interactivos utilizan una interfaz tridimensional brindando un sentido de realidad a diferencia de otros sistemas. Las características de universalidad y accesibilidad de los sistemas interactivos permiten a los usuarios familiarizarse con conceptos abstractos y complejos [Schmitt, B., 2010].

Al investigar el sentido de presencia y la cognición espacial, se descubrió que el uso de sistemas interactivos brindan apoyo a las actividades cognitivas de los diseñadores [Kim, M., 2006].

Diseño centrado en las personas:

Se utiliza el diseño centrado en las personas para el desarrollo de sistemas interactivos.

El diseño centrado en las personas esta dedicado al diseño y desarrollo de sistemas interactivos aplicando factores ergonómicos, técnicas y conocimiento sobre usabilidad, permitiendo que sean más fáciles de utilizar.

Granollers, et al, integran la ingeniería de software con la ingeniería de usabilidad y la HCI para desarrollar el Modelo de Procesos de la Ingeniería de la Usabilidad y de la Accesibilidad llamado MPIu+a.

En el proceso de desarrollo de un sistema interactivo, se debe dar importancia a la etapa de diseño. Se destaca la necesidad de adoptar estándares para el diseño de sistemas interactivos [Granollers, et al, 2004].

El diseño es uno de los criterios más importantes para la evaluación y acreditación en los programas de ingeniería. El enfoque de diseño centrado en las personas contribuye a la innovación en diseño de ingeniería y ha mostrado además incrementar la productividad, mejorar la calidad, reducir errores y reducir costos de producción. En la globalmente

competitiva economía, es más importante que nunca el desarrollar habilidades de diseño efectivas en los estudiantes. Se necesita entender las maneras en que los estudiantes comprenden y experimentan el diseño centrado en las personas [Zoltowski, C., 2012].

ISO 9241-910:2010:

Esta serie normativa se enfoca en la calidad de la usabilidad y accesibilidad, así como a formalizar los aspectos cualitativos de HCI.

Constituye un marco de trabajo para el diseño centrado en las personas al integrar diferentes procesos de diseño y desarrollo apropiados a un contexto en particular; complementando las tecnologías de diseño existentes.

El éxito o fracaso de un nuevo sistema interactivo depende en gran parte de la correcta definición de los requisitos.

Los requisitos de los sistemas interactivos suelen estar enfocados en el qué debe hacer el sistema y no en el cómo debe hacerlo, por lo que suelen clasificarse en requisitos funcionales y no funcionales.

1.3 Objetivos

1.3.1. Objetivo general

Desarrollar aplicaciones interactivas basadas en Phidgets para su uso en el apoyo a la enseñanza de conceptos introductorios en las disciplinas de computación y electrónica.

1.3.2. Objetivos específicos

- Caracterizar los diferentes dispositivos Phidgets.
- Identificar los requerimientos para la construcción de dispositivos adicionales que complementen o sirvan de reemplazo para los Phidgets existentes.
- Proponer estrategias y recomendaciones para el uso de Phidgets.
- Diseñar y construir sistemas interactivos que represente conceptos introductorios de las áreas de computación y electrónica.

1.4 Metodología

Primeramente, se busca y revisa información bibliográfica referente al uso de sistemas interactivos como apoyo a la enseñanza de la ciencia.

Se revisan las características de los sistemas de computo empujado, en particular de los Phidgets.

Se busca y se comparan trabajos relacionados a Phidgets.

Se prueba la funcionalidad de los diferentes dispositivos Phidgets y sus herramientas de desarrollo.

Se identifican los requerimientos para la construcción de dispositivos adicionales que sirvan de reemplazo para los Phidgets.

Se define sistemas interactivos a ser diseñados y desarrollados.

Se diseñan, desarrollan y prueban sistemas interactivos basados en Phidgets.

Finalmente, se elabora una propuesta para uso de los Phidgets en contextos educativos, de acuerdo a los contenidos temáticos propuestos para las disciplinas de computación y electrónica.

1.5 Esquema general de la Tesis

Después de la introducción, en el capítulo dos se presenta definición y caracterización de los dispositivos Phidgets.

Capítulo 2

Phidgets

En esta sección se presenta una definición de Phidgets. De la misma manera se muestra un preámbulo del origen de ellos, y el impacto que puede llegar a tener en los avances tecnológicos de hoy en día. Por otra parte, se presentan las características de los dispositivos Phidgets, tanto de hardware como de software.

2.1 Definición de Phidgets

Los Phidgets son dispositivos de hardware para entrada y salida de datos. Deben su nombre al hecho de que pueden representar físicamente a los componentes para interfaces gráficas denominados *Widgets*, los cuales son pequeñas aplicaciones de software usadas para dar fácil acceso a funciones utilizadas frecuentemente por el usuario [Greenberg, S., Boyle, M., 2002].

Phidgets surgió de un proyecto de investigación en 2001, dirigido por Saul Greenberg en el Departamento de Ciencias de la Computación. En dicho proyecto se encontró que el mayor obstáculo de los programadores es la dificultad de desarrollar y combinar dispositivos físicos e interactuar con ellos con lenguajes de programación convencionales [Greenberg, S. y Fitchett, C., 2001].

Desde entonces, la compañía Phidgets Inc. desarrolla estos dispositivos empaquetados de una manera tal que facilita su programación. El propósito fundamental de Phidgets es permitir la incorporación de entradas y salidas físicas de diversos tipos en aplicaciones interactivas basadas en computadora [Greenberg, S., Boyle, M., 2002].

2.2 Características de Phidgets

Los Phidgets son versátiles debido a que su manipulación con la computadora es de forma directa, donde a nivel de hardware, solo es necesario conectar el dispositivo al puerto USB. Toda la complejidad que implica el comunicarse con el puerto USB es atendida por el API desarrollado por la compañía Phidgets [Phidgets Inc]. Por otra parte, a nivel de

software las aplicaciones pueden ser desarrolladas rápidamente por los programadores utilizando uno de los lenguajes de programación que existen dentro de la amplia gama de lenguajes compatibles con Phidgets. Además, en el sitio web oficial de la compañía existe una comunidad de usuarios que constantemente actualizan el software, añadiendo cada vez más opciones para utilizarlos.

El controlador que se utiliza se conoce como Phidget21 y se encuentra disponible para los sistemas operativos más usados. Todo esto permite a los desarrolladores utilizar interfaces físicas personalizadas con el propósito de interactuar con aplicaciones convencionales [Greenberg, S., Boyle, M., 2002].

2.3 Hardware

Entre la gama de dispositivos Phidgets, además de los que utilizan como interfaz de comunicación el puerto USB, existen otros módulos que se pueden comunicar de manera remota con una computadora, ya sea por medio de conexión Ethernet o con un adaptador Wi-Fi. A estos dispositivos se les conoce como SBC (*Single Board Computer*).

2.3.1. PhidgetInterfaceKit

En la figura 2.1 se muestra uno de estos dispositivos, el cual se utiliza habitualmente para conectar y manipular diferentes tipos de sensores y actuadores a través de señales de entrada y salida. Estas señales están alimentadas con un voltaje de cinco volts proporcionado por el puerto USB. A través de las entradas digitales, se puede leer información proveniente de sensores de tipo lógico, como botones e interruptores. A través de las salidas digitales, se puede controlar dispositivos lógicos, como LEDs, relevadores, transistores, opto-acopladores, etcétera. Por otra parte, a través de las entradas analógicas, se pueden conectar sensores, como de tipo analógico. Las señales son alimentadas con el voltaje proporcionado por el puerto USB, lo cual brinda la oportunidad de utilizar otro tipo de sensores no exclusivos de la compañía Phidgets, mientras que acepten señales de cinco volts.

Existen varios dispositivos de este tipo. En todos ellos, su interacción con la computadora es a través del puerto USB. Y su función principal es servir como interfaz

entre una computadora y otros sensores de la misma compañía. Varían en la cantidad y tipo de salidas que pueden tener. El más sencillo de ellos cuenta con dos de cada tipo de entradas y salidas. Por otra parte, uno de los modelos está especializado en solo entradas y salidas digitales, con dieciséis de cada una. Cada uno de ellos es controlado por un microcontrolador, el cual adapta la señal para ser leída y manipulada como información digital por la computadora. El microcontrolador, empotrado en el dispositivo *PhidgetInterfaceKit 8/8/8* es del tipo MCP23S17. Este microcontrolador es fabricado por la compañía Microchip, cuenta con veintiocho clavijas de las cuales dieciséis son bidireccionales para entrada y salida de datos. Cuenta, además, con una interfaz de comunicación de alta velocidad, bus SPI (*Serial Peripheral Interface*), con un máximo de frecuencia de 10MHz.

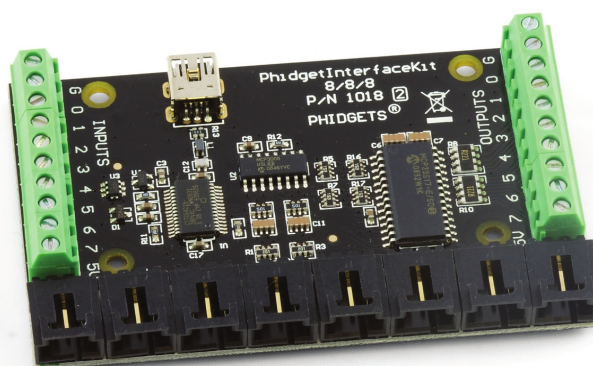


Figura 2.1. PhidgetInterfaceKit 8/8/8, cuenta con ocho entradas analógicas y ocho digitales, además de ocho salidas digitales.¹

2.3.2. Sensores analógicos

Este tipo de sensores regresan al módulo señales que varían de 0 a 5V, dependiendo de las características físicas del sensor. En la tabla 2.1 se muestran algunas de estas características que hacen a los sensores ideales para ser usados con distintos propósitos, dependiendo del problema a solucionar.

La señal analógica es interpretada por la computadora para permitir a los desarrolladores el utilizar, en el código de los programas, valores de cero a mil en números

¹ Imagen extraída del sitio web <http://www.phidgets.com>.

enteros, independientemente del lenguaje de programación [Phidgets Inc]. Dependiendo de la magnitud física que mide cada sensor, el valor de la señal que manda a la computadora debe ser interpretada debidamente por el programador. Para ello se proporcionan formulas de equivalencia en las guías de cada sensor.

Tabla 2.1. Características más importantes de los sensores analógicos.

Sensor	Tipo	Valor en reposo	Valor Activo	Consumo de corriente
Magnético	Ambiente	500	0-1000	2mA
Temperatura	Ambiente	~	0-1000	1mA
Luz	Ambiente	~	0-1000	2mA
Reflexivo IR 5mm	Distancia	1000	< 400	15mA
Reflexivo IR 10cm	Distancia	1000	< 100	9mA-70mA
Rotación	Resistivo	~	0-1000	500μA Max
Barra deslizante	Resistivo	~	0-1000	150μA
Palanca de dos ejes	Resistivo	500, 500	0-1000, 0-1000	1mA
Fuerza	Resistivo	0	> 0	500μA Max
Vibración	Movimiento	500	0-100	400μA
Tacto	Capacitivo	1000	0	150μA

Para conectar los sensores con un módulo interfaz se necesita de conectores de tres hilos de alambre, hembra-hembra. Dos de los hilos alimentan al circuito del sensor mientras el otro manda la señal analógica al microcontrolador del InterfaceKit. En la figura 2.2 se presentan dos sensores de distancia fabricados por la compañía Phidgets.



Figura 2.2. Sensores de distancia infrarrojos de 5mm y 10cm respectivamente.²

Gracias a la manera en que funcionan estos dispositivos, el desarrollador puede concentrarse en la implementación de una aplicación, sin necesidad de tener conocimientos

² Imágenes extraídas del sitio web <http://www.phidgets.com>.

avanzados en el área de la electrónica. Algunos de los sensores son meramente resistivos, donde al variar su resistencia varía el voltaje que es regresado a la interfaz física (Figura 2.3). Este tipo de sensores, en términos eléctricos, funcionan de manera similar a los mencionados anteriormente, pero las diferentes características les permiten ser utilizados en distintos tipos de aplicaciones.



Figura 2.3. Sensor de rotación, barra deslizante y palanca de dos ejes.³

2.3.3. PhidgetRFID

Este dispositivo es un lector de Identificación por Radiofrecuencia (RFID, por sus siglas en inglés) compuesto por una antena (Figura 2.4), un transceptor y un decodificador. Se conecta directamente al puerto USB de una computadora. Este método de identificación se ha hecho realidad gracias a los avances en tecnología de componentes electrónicos, que permiten la alimentación remota de dispositivos; así como a las investigaciones y proyectos que se han realizado para sustituir las diferentes maneras de identificación basadas en papel, tarjetas y otros medios, en los cuales se utilizaban código impreso u escrito. Actualmente se utilizan para monitorear diferentes procesos en la industria. Así como, rastrear mercancía o animales, o identificar individuos para concederles ciertos accesos [Paret, D., 2005].

Estos objetos son identificados con diferentes etiquetas RFID pasivas. Una antena RFID es la encargada de inducir una pequeña corriente eléctrica en estos objetos para extraer de ellos, por radiofrecuencia electromagnética, un número de identificación que se representa en forma hexadecimal en la computadora, con una longitud de 64 bits. El alcance del lector es de 8cm en promedio, dependiendo del objeto pasivo. El lector y las etiquetas pasivas RFID que vende Phidgets Inc. utilizan el protocolo EM4102. Utilizan frecuencias bajas de 125kHz para transmitir la información, por lo que otros identificadores

³ Imágenes extraídas del sitio web <http://www.phidgets.com>.

que utilicen este protocolo pueden ser leídos por la antena PhidgetRFID [Phidgets Inc].

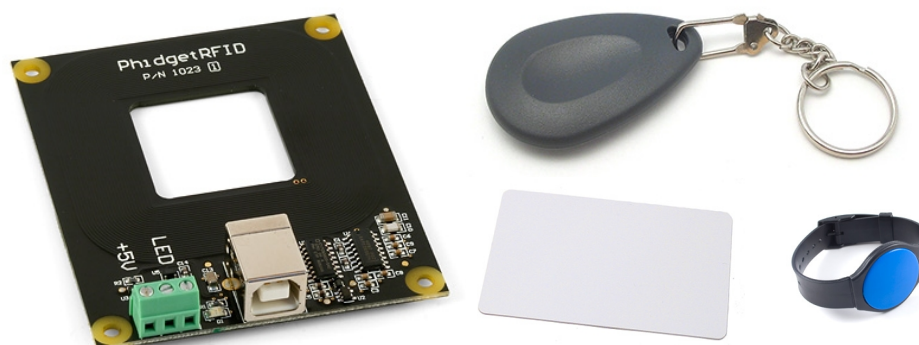


Figura 2.4. Del lado izquierdo se muestra el lector PhidgetRFID y del lado derecho se muestran diferentes tipos de objetos con RFID.⁴

2.3.4. PhidgetSpatial

Estos dispositivos se conectan directamente al puerto USB de la computadora para ser manipulados por los desarrolladores. En la figura 2.5, del lado izquierdo, se muestra un dispositivo que cuentan con acelerómetro, giroscopio y brújula, la cual en realidad es un magnetómetro que reporta la suma de los campos magnéticos que actúan en el dispositivo. Los acelerómetros regresan información a la computadora sobre la aceleración que está sufriendo el Phidget, en tres diferentes ejes. Son capaces de detectar aceleraciones de $\pm 8G$ en múltiplos de la gravedad de la Tierra ($G = 9.81m/s^2$), dependiendo del modelo. El giroscopio informa la velocidad angular que experimenta el dispositivo en tres dimensiones. Si se requiere conocer la posición angular, primero se tiene que calibrar una posición inicial y después se debe integrar la velocidad angular respecto al tiempo.

La aceleración dinámica que se puede detectar con estos dispositivos se puede utilizar de diferentes formas, por ejemplo, para capturar los datos de movimientos en una investigación científica o detectar pequeños movimientos o vibraciones [Phidgets Inc]. Por otra parte, al integrar el valor de la aceleración con respecto al tiempo se puede llegar a encontrar la velocidad y la distancia recorrida del dispositivo. Otra manera de utilizar estos Phidgets es detectando la aceleración estática con lo que se puede calcular la inclinación del dispositivo [Chen, K.Y., Bassett, D.R., Jr., 2005].

⁴ Imágenes extraídas del sitio web <http://www.phidgets.com>.



Figura 2.5. Del lado izquierdo se muestra un Phidget de alta precisión que incluye acelerómetro, giroscopio y brújula. Del lado derecho se muestra un acelerómetro de 3 ejes.⁵

2.3.5. PhidgetAdvancedServo

Estos módulos permiten controlar la posición, velocidad y aceleración de servomotores. La función de los servomotores es posicionar mecánicamente elementos en una determinada posición, en un tiempo establecido y con cierta precisión requerida. Para ello, los servomotores cuentan con un sistema de engranaje conectado a un potenciómetro que regula el ancho de pulso de una señal, a este proceso se le conoce como Modulación por Ancho de Pulso (PWM, por sus siglas en inglés). El utilizar PWM permite regular la velocidad de giro y mantiene el par del motor constante. Por lo regular los servomotores tienen un rango de movimiento de 180° a 360°. Gracias a estas características de control son utilizados con frecuencia para proyectos de robótica. Otro tipo de servomotor son los de rotación continua. Donde la diferencia, con respecto a los otros, es que se reemplaza el potenciómetro por dos resistencias iguales, de esta manera siempre reporta que se encuentra en la posición inicial. A través del control de posición del software se controlan la velocidad y dirección del motor, sin importar la posición [Hegde, G.S., 2010].

Existen dos tipos diferentes de controladores de la marca Phidgets. Uno de ellos solo puede controlar un servomotor y su consumo máximo de corriente no debe de pasar de 450mA ya que se alimenta directamente del voltaje del puerto USB. Igualmente, existe otro controlador con el que se puede manipular hasta ocho servomotores; su consumo máximo es de 3A ya que se alimenta de una fuente de poder externa de 6-15V. Por otra parte, este

⁵ Imágenes extraídas del sitio web <http://www.phidgets.com>.

controlador incluye una salida digital para activar otros dispositivos digitales (Figura 2.6). Este último controlador de servomotores cuenta con una protección de sobre-tensión para los servomotores, gracias a un regulador de conmutación. Una de las bondades de este controlador es que puede medir el consumo actual de cada servomotor por separado [Phidgets Inc].

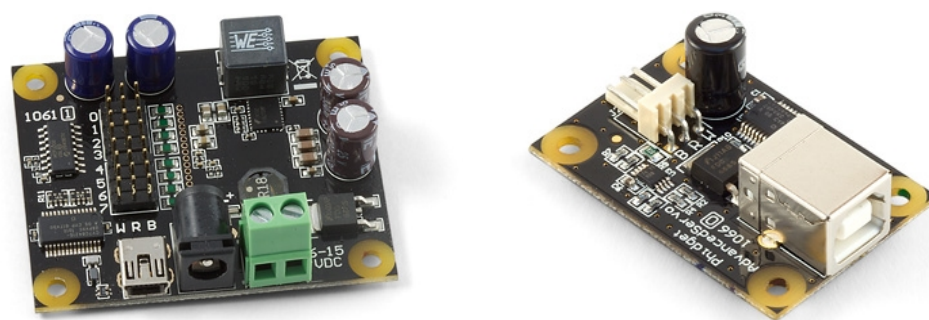


Figura 2.6. Controladores de posición, velocidad y aceleración de servomotores.⁶

2.4 Software

El software necesario para utilizar los dispositivos Phidgets se encuentra disponible en el sitio Web de la compañía. En dicha página se proporcionan hipervínculos de descarga directa para los controladores Phidget21 para los sistemas operativos Windows, Linux, MacOS X, iOS, Windows Mobile, y los *firmware* para los Phidgets SBC, además de guías y consejos de uso e instalación [Phidgets Inc].

En cuanto a lenguajes de programación se refiere, en el mismo sitio web se proporcionan las bibliotecas y ejemplos de código en C/C++, C#, Java, Flash AS3, Matlab, LabView, VisualBasic 6.0, VisualBasic Script, VisualBasic.NET, Phyton, Cocoa, entre otros. Esta interoperabilidad entre sistemas y la amplia gama de lenguajes de programación son una de las mayores ventajas de utilizar Phidgets, ya que permite a los programadores contar con múltiples soluciones de diseño para desarrollar aplicaciones interactivas en el ambiente de programación que más se identifique.

⁶ Imágenes extraídas del sitio web <http://www.phidgets.com>.

Capítulo 3

Desarrollo de aplicaciones con Phidgets

En éste capítulo se describen las actividades a seguir durante el proceso de diseño de sistemas interactivos utilizando Phidgets. En estos sistemas se combinan tanto hardware como software, así como servicios que reciben las entradas de los usuarios y le comunican las salidas del sistema a los mismos. En la sección 3.1 se implementa un proceso de diseño centrado en las personas, el cual tiene sus raíces en los campos de ergonomía, informática y en la inteligencia artificial. Estos campos se aplican en estándares internacionales, como lo es el estándar ISO 9241-210:2010 “*Ergonomics of human-centred system interaction*” [Giacomin, J., 2012]. Dicho proceso se enfoca en el diseño y desarrollo de sistemas que permiten que los sistemas interactivos sean más fáciles de utilizar, debido a que aplican factores ergonómicos, técnicas y conocimientos sobre usabilidad. Este estándar internacional recomienda específicamente seis características que se deben de tomar en cuenta en el proceso de diseño de sistemas interactivos [ISO 9241-210, 2010]:

- Adoptar habilidades y perspectivas multidisciplinarias.
- Entender explícitamente al usuario, las tareas y el entorno.
- Evaluar el diseño centrado en el usuario.
- Considerar toda la experiencia del usuario.
- Involucrar a los usuarios durante el diseño y desarrollo.
- Llevar acabo un proceso iterativo.

3.1 Adaptación de una aplicación para usarse con Phidgets

En esta sección se presenta un ejemplo de como adaptar los dispositivos Phidgets en una aplicación ya desarrollada, la cual está diseñada para ser manipulada por el ratón de la computadora. Dicha aplicación consiste en una adaptación del videojuego clásico *Space Invaders*, creado por Toshihiro Nishikado para la empresa japonesa Taito en 1978. A través de la implementación del juego, se busca generar una experiencia de uso natural, por lo que se incorporan interfaces físicas, adicionales a las típicas de la computadora, como interfaz

de entrada para las acciones que el jugador podrá realizar.

En las siguientes secciones se presentan cuatro actividades principales siguiendo como referencia el diseño centrado en las personas que se propone en la norma ISO 9241-210:2010. La interpretación de estas actividades se muestra en la Figura 3.1.

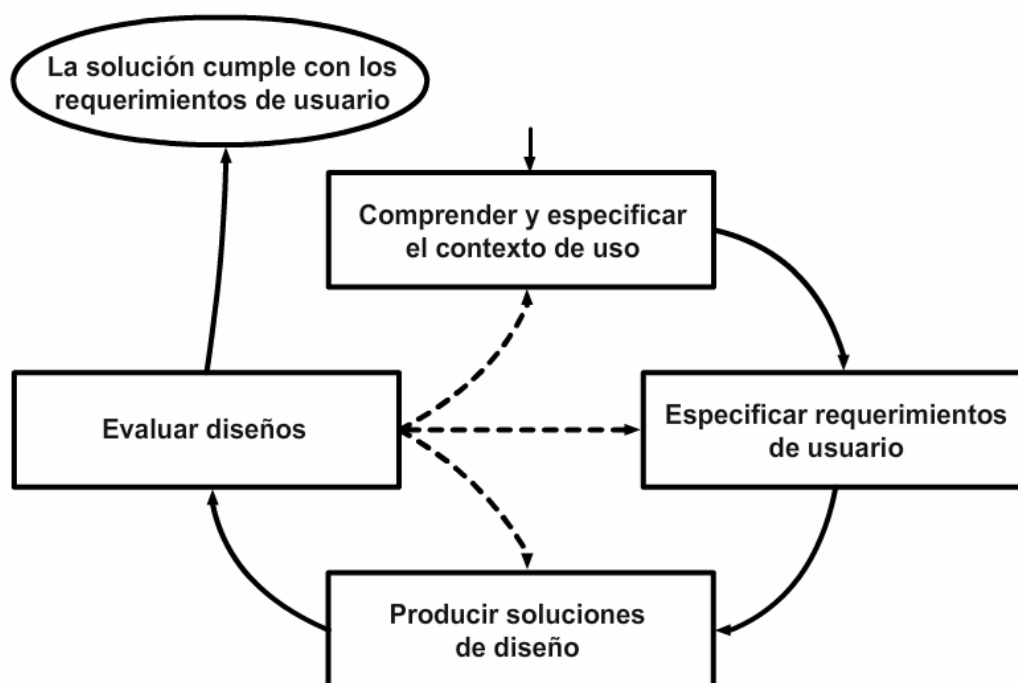


Figura 3.1. Interdependencia de las actividades a realizar en el proceso de diseño centrado en las personas.

1. Comprender y especificar el contexto de uso. Se consideran las diferentes características de los usuarios y las tareas que estos llevarán a cabo. Por otra parte, el ambiente organizacional, técnico y físico son factores importantes que hay que considerar durante el diseño.
2. Especificar requerimientos de usuario. Se identifica lo que el usuario tiene que lograr a través del sistema. De la misma forma, se especifica cualquier limitación impuesta por el contexto de uso.
3. Producir soluciones de diseño. Se plantean las soluciones que puedan cumplir con los requerimientos de usuario. Incluyendo los aspectos físicos, emocionales y

estéticos.

4. Evaluación del diseño. Para obtener un mejor entendimiento de las necesidades del usuario se deben de evaluar los conceptos de diseño en las etapas iniciales, así como en las iteraciones siguientes. Se pueden utilizar pruebas basadas en el usuario complementándolas con evaluaciones basadas en la inspección. También se recomienda un monitoreo a largo plazo.

Estas actividades pueden y deben de realizarse de manera iterativa hasta que la solución cumpla con los requerimientos de usuario.

3.1.1. Comprender y especificar el contexto de uso

La primera actividad, dentro del proceso de diseño que se está siguiendo, corresponde a la definición del contexto en el cual se utilizará el sistema completo.

Al tratarse de un juego de vídeo, el sistema se encuentra orientado principalmente a niños y jóvenes que tengan la habilidad de manipular los dispositivos de entrada propuestos. Se puede inferir además, que la mayor parte de estos usuarios ya cuentan con un nivel de experiencia y conocimiento suficiente en el uso de juegos interactivos del género de disparos en primera y tercera persona.

3.1.2. Especificar requerimientos de usuario

El juego *Space Invaders* consiste en controlar un objeto que representa una nave espacial. Dicha nave debe evitar que un grupo de objetos, representado por naves extraterrestres, se acerquen a la Tierra. Para lograrlo, la nave puede desplazarse en la pantalla sobre el eje de las x y disparar a las naves extraterrestres, al tiempo que debe evitar los disparos que le son lanzados.

El objetivo de los usuarios al jugar *Space Invaders* es utilizar su destreza y habilidades para avanzar por los diferentes niveles de dificultad utilizando un dispositivo físico, propuesto anteriormente, como interfaz de entrada. Durante la utilización del juego los usuarios incrementarán su conocimiento tácito de manera natural al tener retroalimentación sobre su desempeño y al aprender reglas y técnicas para seguir jugando.

En la versión del juego de la que se parte para el desarrollo de este sistema

interactivo, el control del desplazamiento y el disparo se realizan a través del ratón de la computadora. Este es un dispositivo que cuenta con una interfaz de uso bastante simple. Sin embargo, su diseño resulta más apropiado para las denominadas interfaces WIMP (por sus siglas en inglés de Windows-Icons-Menus-Pointer o ventanas-íconos-menús-apuntador), mismas que no suelen ser las utilizadas dentro de un juego de vídeo [Wigdor, D., et al, 2011]. Además, varios estudios muestran que el uso del ratón no suele ser rápido de aprender ni de dominar entre usuarios de computadora muy jóvenes o que cuentan con alguna discapacidad física [Wu, T.F., et al, 2010], [Girard, S., et al, 2009].

De manera adicional al desarrollo del juego como tal, se agrega el requerimiento de generar en el usuario una experiencia de uso natural al proporcionarle interfaces adicionales a las típicas de la computadora, como son el ratón y teclado. En este sentido, se busca que las acciones requeridas por el sistema correspondan a acciones realizadas por el usuario. Esto puede lograrse por medio de dispositivos físicos incorporados a objetos que, al minimizar el periodo de aprendizaje de uso y dominio de la interfaz, no impongan una carga cognitiva adicional en la persona.

De esta forma, el requerimiento principal queda definido como la adaptación del juego de vídeo *Space Invaders* utilizando una interfaz de entrada física adicional a las convencionales de la computadora.

3.1.3. Producir soluciones de diseño

La versión inicial del juego *Space Invaders* que se está implementando se encuentra desarrollada en Flash, utilizando el lenguaje de programación *ActionScript 3.0*. A través de varias iteraciones, de acuerdo al proceso de diseño centrado en las personas, se logra migrar la aplicación del vídeo juego a un sistema interactivo. Dentro de este sistema interactivo, se considera el uso de interfaces de entrada físicas adicionales al ratón, proporcionando al usuario un esquema de interacción basado en interfaces multimodales.

De acuerdo al trabajo presentado por Srikulwong, la multimodalidad se refiere a permitir al usuario el uso de múltiples vías para llevar a cabo la entrada y salida de información [Srikulwong, M., 2007]. Estas variantes ofrecen opciones para la interacción que deben ayudar y extender, de manera apropiada, la forma en la que los usuarios se

comunican y desempeñan sus actividades en diferentes situaciones.

3.1.3.1. Adaptación de la interfaz de entrada

Existen diversas opciones para la implementación de interfaces físicas para la computadora. Unas de las más versátiles son los Phidgets, anteriormente mencionados, debido a que estos dispositivos permiten la incorporación de entradas y salidas físicas de diversos tipos en aplicaciones interactivas basadas en computadora. Los Phidgets proporcionan a los desarrolladores dispositivos físicos lo suficientemente sencillos de utilizar, modificar o combinar para construir una interfaz física de usuario, ocultando los detalles de bajo nivel relativos a su construcción, implementación y comunicación con los puertos de la computadora.

Para esta aplicación se usó la biblioteca propia del lenguaje *ActionScript 3.0*, el software controlador *Phidget21* y el paquete *com.phidget*. El cual debe incluirse en la raíz del proyecto Flash. Adicionalmente, para utilizar Phidgets con *ActionScript 3.0*, se requiere el uso del Phidget *Web Service*, mismo que es proporcionado junto con el controlador.

La interfaz física para el juego fue paulatinamente modificada; cada adaptación se dio a través de 4 iteraciones. De igual forma, se fue agregando el código necesario para usar Phidgets, así como elementos multimedia que complementaran la experiencia de uso. El progreso de estas iteraciones se presenta en la Tabla 3.1.

Tabla 3.1. Muestra las características de las cuatro iteraciones realizadas.

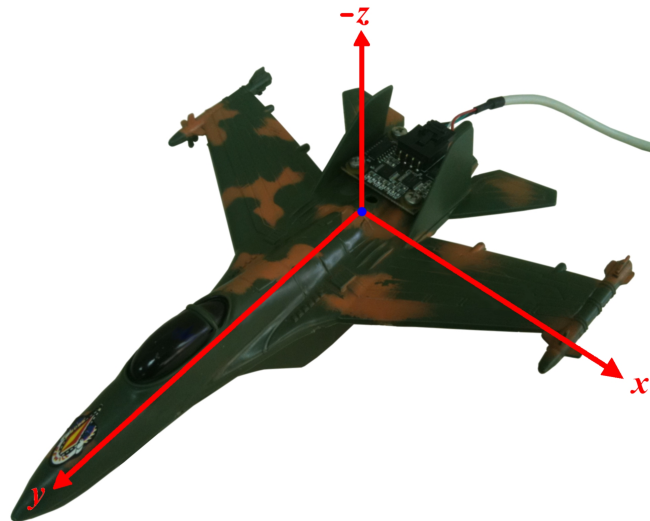
Iteración	Característica
Primera	- Modificar la respuesta del objeto <i>HeroShip</i> para controlarlo con el <i>PhidgetAccelerometer 3-Axis</i> . - Integrar el Phidget en un avión de cartón.
Segunda	- Agregar sonido y definir imágenes específicas para los objetos <i>HeroShip</i> , <i>AlienShip</i> , <i>HeroCannonBall</i> , <i>AlienCannonBall</i> y <i>AlienMine</i> . - Integrar el Phidget en un avión de plástico.
Tercera	- Agregar complejidad de velocidad y cantidad de objetos <i>AlienShip</i> , y velocidades y frecuencia de los objetos <i>AlienCannonBall</i> y <i>AlienMine</i> .
Cuarta	- Agregar al avión un disparador para atacar sin tener que estar al pendiente de la

Tabla 3.1. Muestra las características de las cuatro iteraciones realizadas.

	aceleración dinámica de los ejes y y z .
	- Agregar la función en la clase <i>HeroShip</i> .

Como se observa en la Tabla 3.1, desde la primera iteración se utilizó un dispositivo de entrada llamado *PhidgetAccelerometer 3-Axis*. Este es un sensor de aceleración que, en función del valor de la aceleración medido en cada uno de sus ejes, permite determinar el tipo de movimiento al que se le está sometiendo.

Para utilizar el Phidget con acelerómetro en la aplicación de *Space Invaders*, se montó en un avión de plástico. El avión físico se utiliza para representar a una nave virtual de la aplicación. En la Figura 3.2 se observa el avión con el *PhidgetAccelerometer 3-Axis* empotrado sobre la parte superior trasera del mismo. En la misma figura se muestran los ejes x , y y z que mide el acelerómetro. Al mover el avión y colocarlo en diferentes posiciones, el acelerómetro entrega valores de aceleración con los cuales permite a la aplicación en la computadora determinar la dirección de movimiento y aplicarla al juego para desplazarse horizontalmente en el escenario.

**Figura 3.2.** Acelerómetro de la marca Phidget empotrado en un avión de plástico.

Las primeras dos iteraciones obedecen principalmente a aspectos considerados desde la especificación de los requerimientos para la aplicación. Las modificaciones y adaptaciones realizadas en la tercera y cuarta iteración, se definieron a partir de unas evaluaciones y pruebas llevadas a cabo con usuarios. Los resultados y el análisis de este proceso de pruebas se muestran en el capítulo siguiente.

3.1.3.2. Descripción del sistema interactivo *Space Invaders*

Para describir el funcionamiento del sistema interactivo *Space Invaders*, se optó por utilizar un método de representación llamado Gráfica Rica (*Rich Picture*).

Con el fin de ayudar a comprender claramente cómo el sistema funciona, se utiliza este método. Ya que con él se pueden representar de manera gráfica: situaciones, problemas o conceptos abstractos. Para una clara comunicación entre el diseñador y los clientes, esta es la herramienta más flexible y universal que se puede conseguir. No tiene reglas ni limitaciones a la hora de plasmar las ideas. Gracias a que las Gráficas Ricas cuentan con características de universalidad y son simples de entender, las hace un candidato ideal para que cualquier interesado las pueda utilizar [Horan, P., 2000].

Pat Horan encontró algunas de las ventajas de utilizar las Gráficas Ricas, estas son incluidas en su trabajo de la Universidad La Trobe, de la División de Tecnologías de la Información. Por una parte encontró que la mayoría de los estudiantes prefieren las imágenes que el texto, ya que pueden absorber más información de una imagen de un tema complejo que de una descripción textual. Además se requiere poca habilidad artística, una imagen poco profesional, una vez que se toma en serio, invita a los no expertos a participar, ya sea en el diseño o en la interpretación. Es independiente del lenguaje. Esto puede ser de mucha ayuda para el diseñador como para los que verán la imagen. Se puede representar, con dibujos, desde una pequeña función hasta un sistema completo. Es modificable, en cualquier parte de la imagen se puede, fácilmente, agregar elementos nuevos o corregir ya existentes. Proveen de las bases para la comunicación y la negociación.

En la Figura 3.3 se representan los íconos utilizados en la gráfica rica para

representar el sistema interactivo *Space Invaders*.

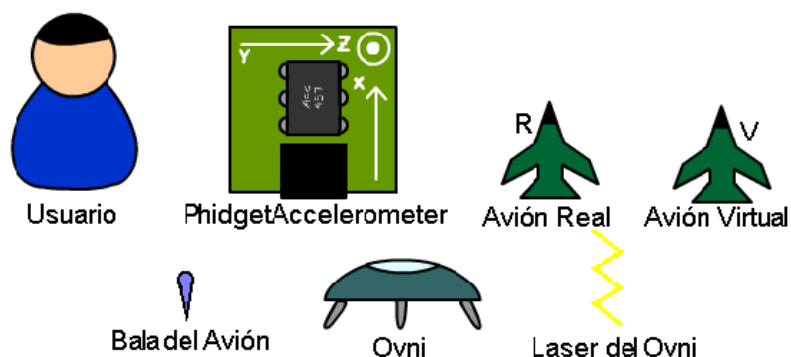


Figura 3.3. Imágenes representativas del sistema interactivo para la gráfica rica.

Para representar las diferentes actividades se muestran tres diferentes dibujos. La Figura 3.4 indica el resultado de empotrar el acelerómetro en el avión de plástico.

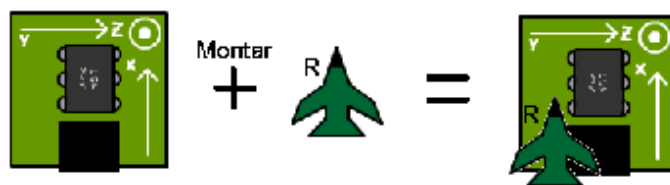


Figura 3.4. Gráfica rica del avión real con un acelerómetro *Phidget* empotrado.

Por otra parte en la Figura 3.5 se representan tres maneras en que el usuario puede manipular el avión con acelerómetro. Girarlo a la izquierda hará que el avión virtual se desplace a la izquierda del escenario; análogamente para el sentido contrario. Además se aprecia que al efectuar, el usuario, un movimiento hacia atrás y hacia enfrente sobre el avión real, el avión virtual realizará un disparo.

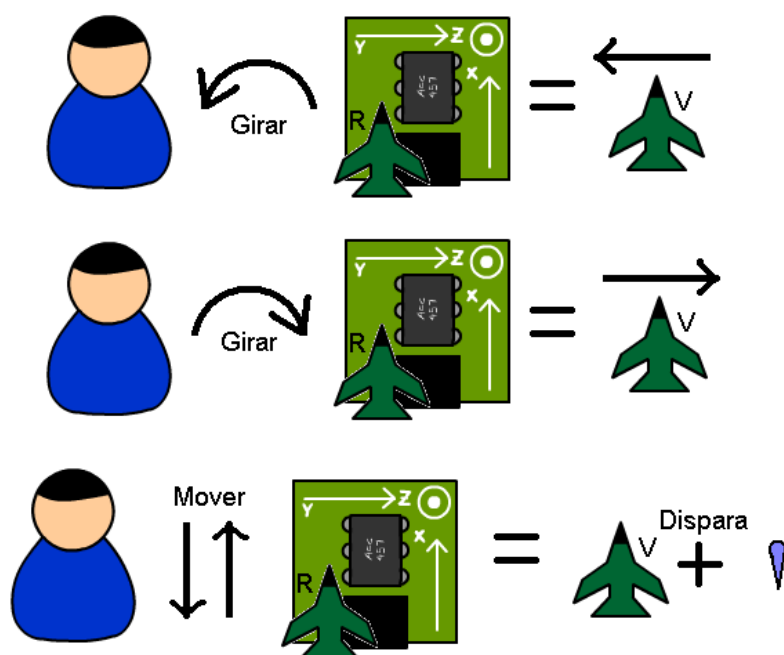


Figura 3.5. Gráfica rica de las acciones que puede realizar el usuario.

En la Figura 3.6 se muestra las interacciones de los objetos virtuales. Donde, si la bala del avión hace contacto con un ovni, el ovni se destruye. Además, el ovni lanzará su láser una vez en un tiempo aleatorio. Si el láser de algún ovni hace contacto con el avión virtual, este se destruye.

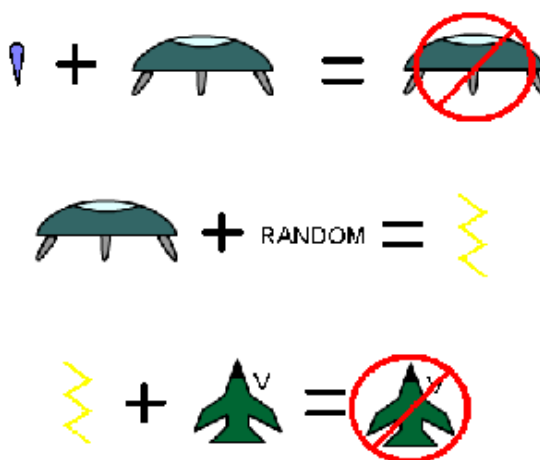


Figura 3.6. Gráfica rica que muestra la interacción de los elementos del juego.

3.1.3.3. Adaptación del código

Los recursos utilizados durante el desarrollo de la aplicación comprenden catorce clases en el lenguaje de programación *ActionScript 3.0*. Los cuales se dividen en los paquetes de: Naves (*Ships*), Armamento (*Weapon*) y Jugador (*Gamer*).

En la clase *HeroShip.as* se utilizó una instancia de la clase *PhidgetAccelerometer* para agregar en ella un escucha de eventos de tipo *PhidgetDataEvent*, el cual al detectar cualquier cambio en la aceleración mandará a ejecutar las funciones *doMoveShip()* y *doFire()*. El siguiente código describe las funciones anteriormente mencionadas. El primer fragmento de código de la función *doMoveShip()*, muestra un código para rotar la nave virtual *HeroShip*, y otro que permite desplazar a la nave a la izquierda cuando el valor de la aceleración estática en el eje *x* es negativo; a la derecha en caso contrario. Mientras que el código de la función *doFire()* se utiliza para hacer que la nave dispare un misil a la vez por el escenario cuando el valor absoluto de la aceleración dinámica es, en cualquiera de los ejes *y* o *z*, mayor a 1.8G.

Código 3.1.

```
protected function doMoveShip(event:PhidgetDataEvent):void
{
    if(event.Data < 1)
        if(event.Index==0){
```

Código 3.2. Rotación de la nave.

```
this.rotation = -Number(event.Data)*30;
this.scaleX = 1 - 0.5 * Math.abs(Number(event.Data));
```

Código 3.3. Desplazamiento de la nave.

```
if(Math.abs(Number(event.Data) > 0.25)
    this.x -= 9*Number(event.Data);
}
if(this.x < 0)
    this.x = 10;
if(this.x > stage.stageWidth)
    this.x = stage.stageWidth-10;
}
```

Código 3.4.

```
protected function doFire(event:PhidgetDataEvent:void
{if(event.Index != 0)
    if(event.Data > 1.8)
        if(!weapon.heroShootInAir){
            cannonSound.play();
```

Código 3.5. Disparo de la nave.

```
        weapon.fire(HeroWeapon.CANNON, stage, this.x, this.y - 25,
        this.parent as Player);
    }
}
```

Para permitir la manipulación de la instancia de la clase *PhidgetAccelerometer*, se necesita primero utilizar una función propia de Phidgets llamada *open()*. Esta función da acceso a los servicios que proporcionan los Phidgets. Se requiere primeramente especificar los valores del servidor y el número de puerto para acceder como servicio Web. El evento

ACCELERATION_CHANGE es activado por el hardware *PhidgetAccelerometer 3-Axis* al suceder cambios de gravedad en cualquiera de sus tres ejes x , y y z . En la clase *HeroShip.as* se puede usar información de este evento. La sintaxis *event.Index* proporciona el eje donde hay cambios, mientras que con *event.Data* se puede obtener el valor de la gravedad en ese eje en un rango numérico de -3 a 3 (el rango depende del modelo del acelerómetro), que representan el valor de la gravedad.

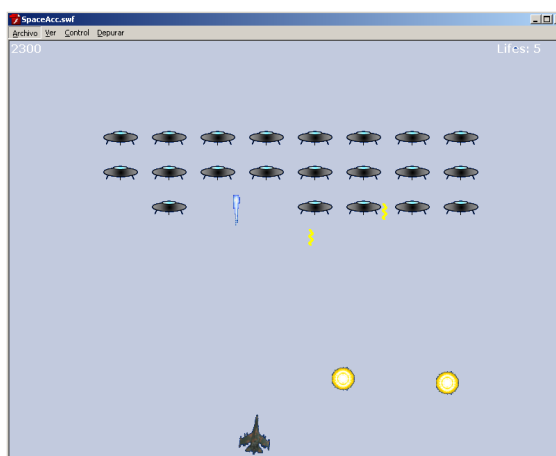


Figura 3.7. Interfaz gráfica del juego *Space Invaders* en funcionamiento.

La interfaz gráfica final del juego se presenta en la Figura 3.7. En ella se observa una representación digital del avión físico realizando un desplazamiento sobre el eje x . Tal efecto es generado debido al procesamiento de los datos entregados por la interfaz física al ocurrir un cambio en el valor de la gravedad en el eje x del acelerómetro.

3.1.4. Evaluación de diseño

Con la primer evaluación de diseño del sistema interactivo, se estableció el asignar imágenes y sonidos para cada objeto en particular. Por otra parte, a partir de este punto se montó el *PhidgetAccelerometer 3-Axis* en un avión de plástico, ya que el prototipo anterior era un avión de cartón.

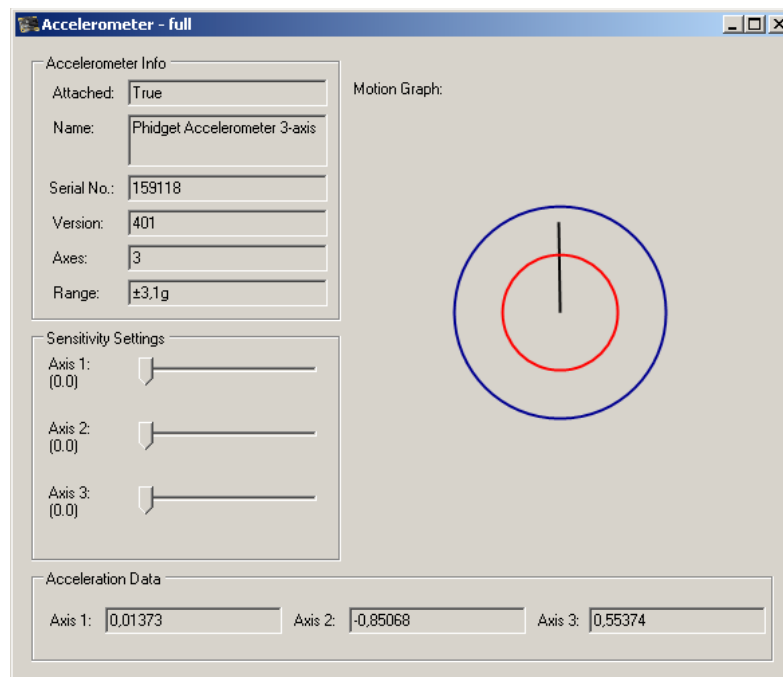


Figura 3.8. Interfaz gráfica para interactuar con el acelerómetro Phidgets.

Para lograr representar los movimientos del avión, de manera adecuada, en el sistema interactivo. En una primera etapa se procedió a caracterizar la respuesta del sensor aplicando diferentes tipos de movimiento. Dicho proceso fue realizado gracias a la ayuda de las herramientas proporcionadas por el controlador y las bibliotecas de Phidgets. En la Figura 3.8 se muestra la interfaz gráfica usada para encontrar los valores de respuesta del acelerómetro.

Con el fin de recopilar información acerca de la usabilidad de la aplicación, se realizaron dos pruebas de valoración a un grupo reducido de estudiantes de cuarto a sexto semestre de las carreras de Licenciado en Sistemas Computacionales e Ingeniero en Computación. En la primera de ellas se utilizó la técnica de pensar en voz alta mientras usaban el sistema interactivo. En la cual se analizaron sus comentarios acerca de la experiencia con el uso de Phidgets. Posteriormente, se aplicaron encuestas personales de satisfacción para conocer las fortalezas y debilidades del funcionamiento de la interfaz física.

3.2 Diseño de una aplicación basada en Phidgets

En esta sección se describe el desarrollo de una aplicación considerando el uso de Phidgets desde el diseño de la misma. De esta manera se busca explorar un enfoque distinto de desarrollo. La aplicación interactiva es un juego didáctico de aritmética para desarrollar la habilidad matemática del usuario. Se decidió utilizar el lenguaje de programación ActionScript 3.0. En dicha aplicación se muestra una ecuación matemática teniendo como incógnita el operador, mostrando los dos operandos y el resultado. De esta manera el usuario tendrá que responder cuál es operador desconocido. Para la interacción en el juego se planteó utilizar etiquetas *RFID* empotradas en cuatro tarjetas, en las cuales cada una representa una operación básica. La aplicación utiliza la antena *RFID* para interpretar las etiquetas y responder a cada una de las ecuaciones.

3.2.1. Descripción de la aplicación con Gráficas Ricas

Para proporcionar un esquema de interacción basado en interfaces multimodales, se dejó a disposición del usuario el uso del teclado y del ratón para su conveniencia. De tal manera que acercar la tarjeta *RFID* de suma, provoca la misma acción que utilizar la tecla correspondiente de suma o seleccionar con el cursor el símbolo de suma. Para describir con mayor detalle la interacción del juego se utilizan gráficas ricas. En la Figura 3.9 se muestran los íconos representativos usados en la gráfica rica. En dicha figura se muestra, en el primer renglón de imágenes, la representación de: el usuario, una ventana que representa la interfaz gráfica de la aplicación, la antena *RFID*, los cuatro operadores básicos y la instrucción de actuar. En el siguiente renglón se pueden apreciar las imágenes representativas de: el creador de ecuaciones aleatorias representado por un cerebro con números y operadores, una ecuación aleatoria, un signo de interrogación que representa el operador incógnito, y las imágenes de una marca de aprobación y otra de error. En el tercer renglón se muestran las imágenes de: un cronómetro representando el tiempo límite para contestar cada ecuación, un marcador en forma de barra representando la vida y otro representando la experiencia, la palabra *Score* representa el puntaje del jugador y la abreviatura *Lv* representa el nivel de dificultad. Los últimos tres íconos representan el tipo de interacción que se realiza, las cuales pueden ser el ratón de la computadora, el teclado o

las tarjetas RFID.

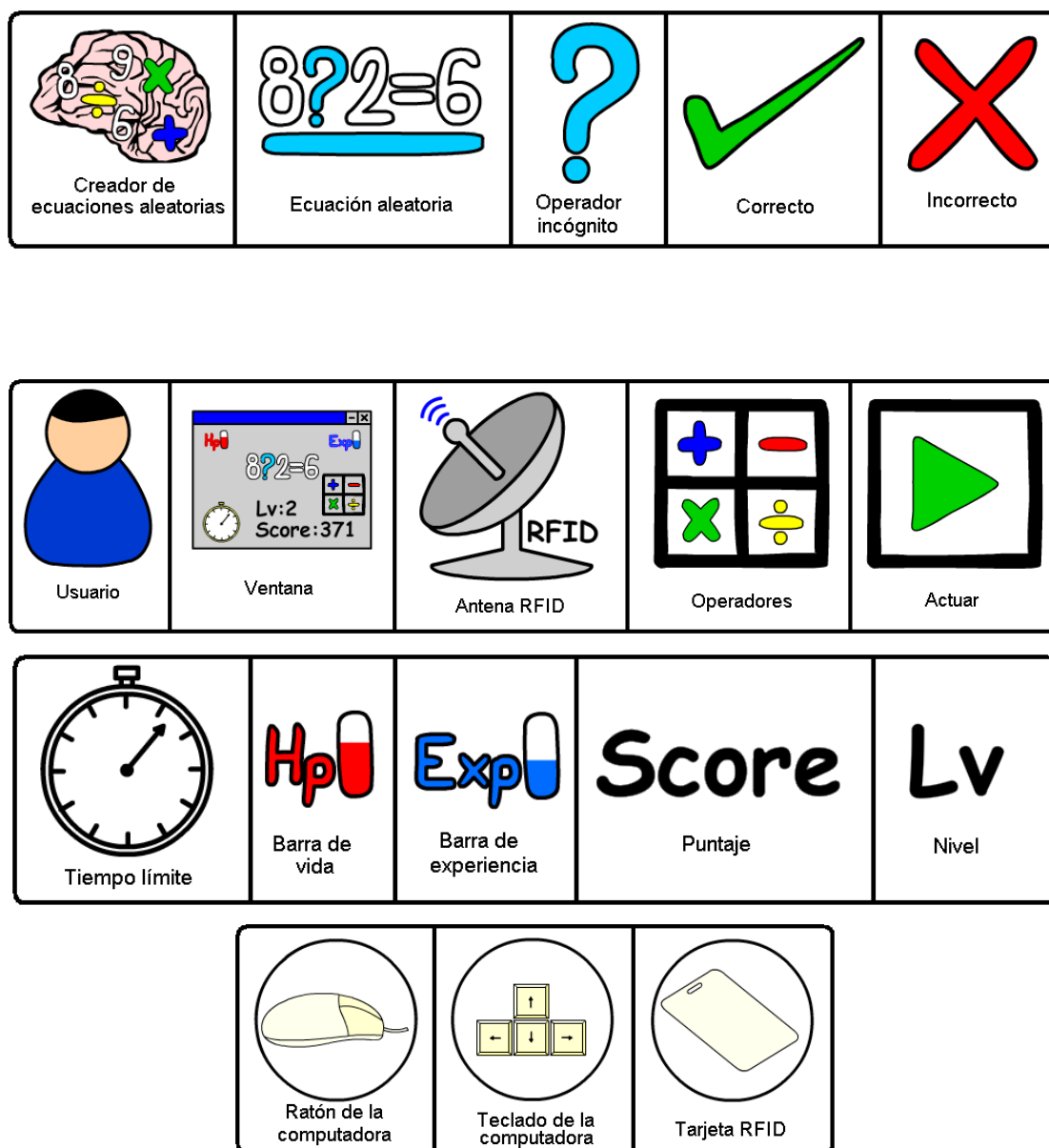


Figura 3.9 Íconos representativos de la gráfica rica.

Para describir la mayor parte de la interacción entre el usuario y la aplicación se presenta la Figura 3.10. Empezando con el usuario seleccionando la opción de iniciar el juego, independientemente de la modalidad de interacción, el juego inicia solicitando una ecuación nueva a la fábrica de ecuaciones. Esta crea una ecuación nueva, la cual se muestra en la interfaz gráfica; además inicia un temporizador que indica el tiempo límite que tiene

el usuario para contestar. Si el tiempo se agota se muestra un error en la pantalla provocando una disminución en la vida del jugador. Lo mismo ocurre si el usuario responde de manera incorrecta la ecuación. Por otra parte, si el jugador contesta correctamente, la vida del jugador aumenta un poco, así como su experiencia y puntuación. Independientemente de la respuesta, la aplicación actualiza la pantalla y solicita otra ecuación.

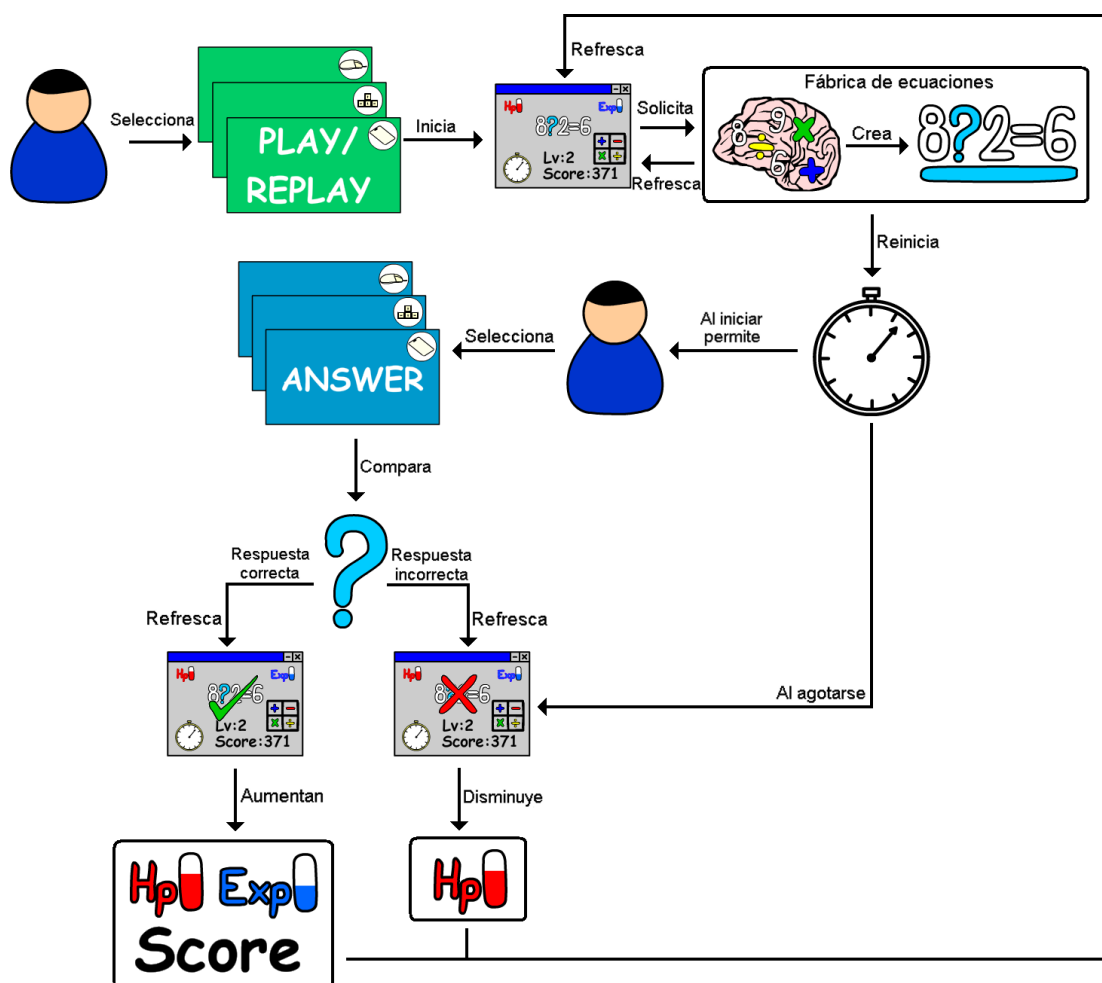


Figura 3.10. Gráfica rica que representa la interacción del usuario con el sistema.

En la Figura 3.11 se muestran las diferentes formas en las que el usuario puede interactuar con el sistema, del lado izquierdo se aprecian las tres modalidades de contestar con un operador, ya sea presionando con el ratón sobre un botón en la interfaz gráfica,

utilizando el teclado (En este caso se propone el uso de las cuatro direcciones del teclado por ser teclas adyacentes entre si. Arriba equivale al operador de suma, abajo al de resta que es el operador inverso de suma, análogamente la derecha es multiplicar e izquierda dividir), o colocando cerca de la antena *RFID* una de las tarjetas marcadas con cada operador aritmético. Por otra parte, del lado derecho de la Figura 3.11 se representan las diferentes modalidades de realizar la interacción de actuar, que puede ser bien un botón gráfico en la pantalla, la tecla de espacio del teclado, o una tarjeta marcada con un triángulo apuntando a la derecha.

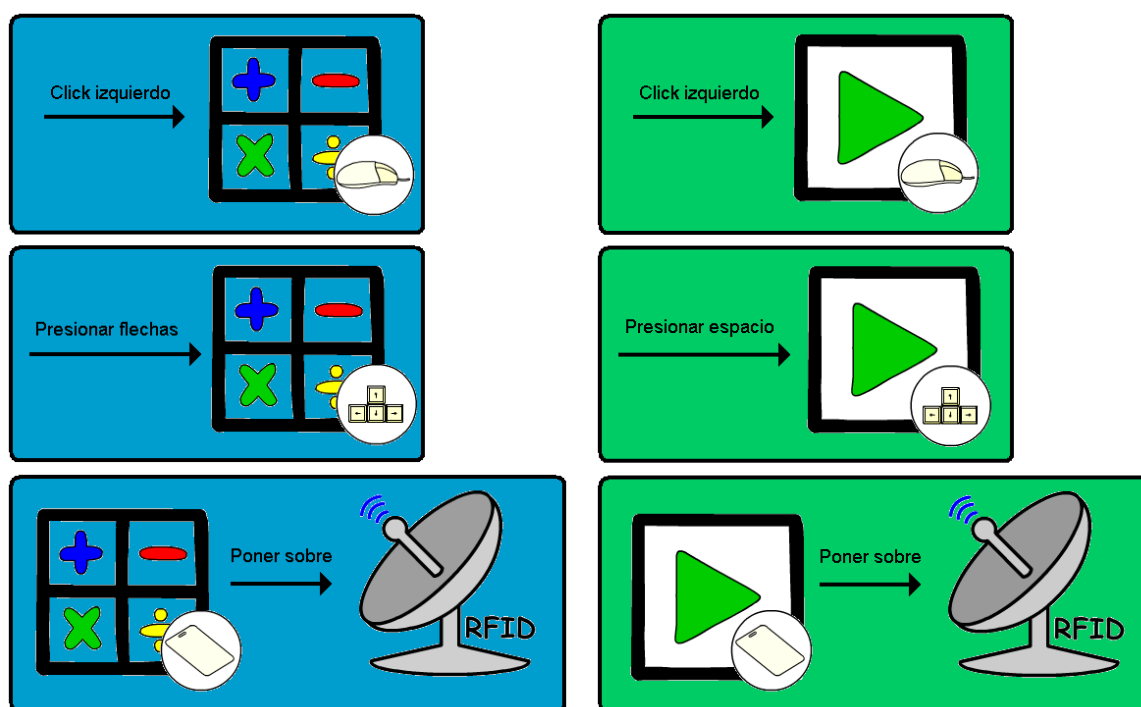


Figura 3.11. Gráficas ricas representativas de las diferentes modalidades que tiene el usuario para interactuar con el sistema.

Para representar otras acciones importantes que ocurren dentro del sistema se muestra la Figura 3.12. Donde, en la parte de arriba, se indica que cuando el marcador de vida del jugador queda vacío se muestra en la interfaz gráfica un mensaje de terminación, de la misma manera se detiene el temporizador. En la parte de abajo de la Figura 3.12 se representa lo que ocurre ir subiendo de nivel. Primeramente, para subir de nivel, el jugador

tiene que llenar la barra de experiencia. Al subir de nivel la barra de experiencia se regresa a cero, la velocidad del temporizador aumenta, de la misma manera que aumenta la dificultad de las ecuaciones. A medida que el nivel del juego aumenta, los números en las ecuaciones aleatorias son más grandes y la frecuencia de números negativos aumenta.

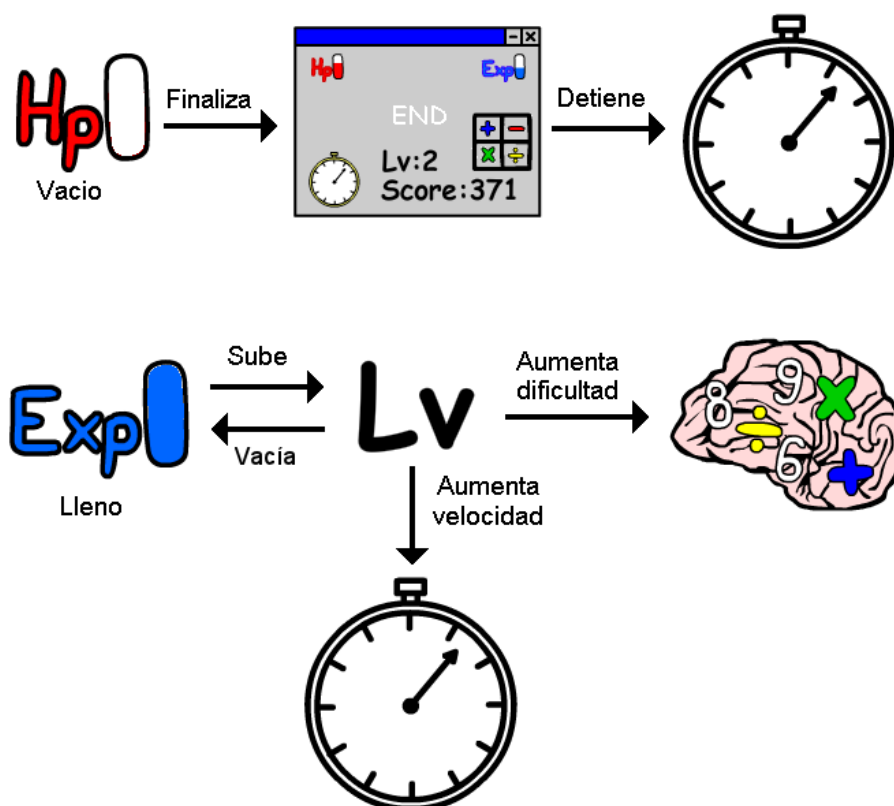


Figura 3.12. Gráfica rica que representa lo que ocurre en el sistema en diferentes estados de las barras de vida y experiencia.

3.3 Uso de la biblioteca Phidget

En esta sección se presenta una serie de pasos a seguir para utilizar la biblioteca Phidget en un proceso de desarrollo. Independientemente del lenguaje de programación en que se esté desarrollando la aplicación, primero se tiene que contar, en la computadora, con el controlador Phidget21. Este controlador se encuentra disponible para descargar en la la

página oficial de Phidgets, en la sección de controladores. Dependiendo del sistema operativo en el que se está trabajando, se descarga la versión correspondiente. Una vez descargado el archivo que contiene el controlador, este se instala. Para el sistema operativo Windows, además de instalar las bibliotecas, se instala un panel de control para administrar los Phidgets conectados a la computadora. Lo siguiente a descargar son las bibliotecas del lenguaje de programación con el que se desarrollará, se encuentra en el apartado de programación. Ahí se encontrarán los archivos comprimidos de las biblioteca de clases y los eventos del lenguaje de programación seleccionado. En estos archivos comprimidos, se incluyen ejemplos de código para utilizar cada dispositivo Phidget.

3.3.1 Ejemplo de uso de una biblioteca Phidget

En este apartado se ejemplifica el uso de las bibliotecas de Phidgets en el ambiente de desarrollo Flash. Los ejemplos proporcionados por la página Web para este ambiente están escritos usando ActionScript3.0 bajo Flash CS3, también conocido como Flash 9, por lo tanto se asume su uso. Otros ambientes de desarrollo que soporten ActionScript3.0 pueden funcionar.

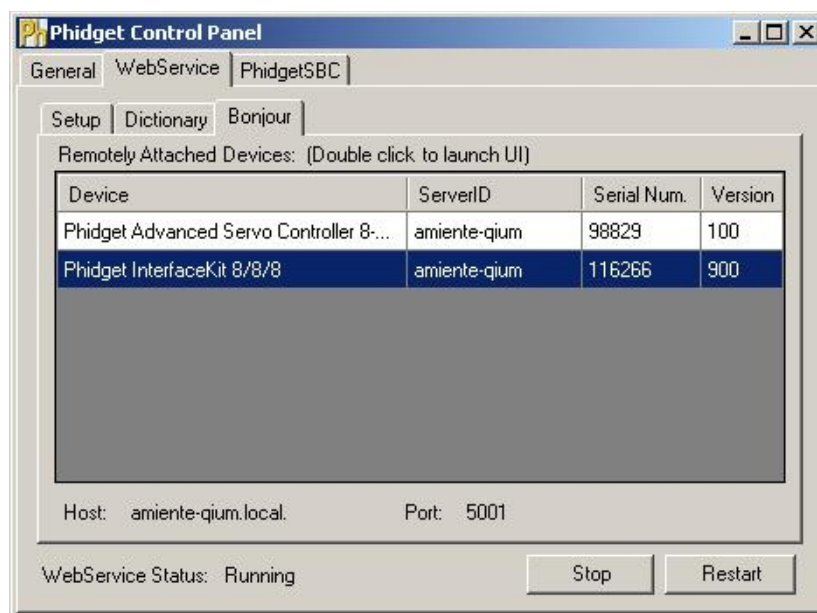


Figura 3.13. Panel de control de Phidgets que muestra el contenido de la Pestaña Bonjour.

Primero hay que asegurarse que el Phidget que se va a utilizar se encuentre correctamente conectado al puerto USB. Después hay que revisar, en el panel de control de Phidgets, que estén en funcionamiento los servicios Web. Esto se puede corroborar en una pestaña del controlador, etiquetada como Bonjour (ver Figura 3.13). Bonjour es una tecnología de detección de redes desarrollada por Apple, Inc. Permite a las computadoras e impresoras comunicarse unas con otras a través de un nuevo protocolo. En este caso los servicios Web son requeridos para poder utilizar los dispositivos Phidgets en las aplicaciones desarrolladas en Flash. Sin embargo no en todos los lenguajes de programación son requeridos los servicios Web.

Posteriormente se abre el programa de la plataforma Flash y se crea un nuevo archivo Flash con el nombre de PhidgetTest. El siguiente paso es copiar la carpeta llamada “com”, que se encuentra en los ejemplos descargados, para pegarla en la dirección raíz del proyecto, lo cual permite utilizar las bibliotecas de Phidgets. En las acciones del primer fotograma se incluyen las bibliotecas del dispositivo Phidget que se está utilizando junto con los eventos:

```
import com.phidgets.PhidgetInterfaceKit;
import com.phidgets.events.*;
```

A continuación se declara el objeto *Phidget* que se va a utilizar y se inicializa. En este ejemplo se declara un objeto *phid* tipo *PhidgetInterfaceKit*:

```
public var phid:PhidgetInterfaceKit;
phid = new PhidgetInterfaceKit();
```

Para conectar el Phidget en el programa se utiliza el método *open* el cual tratará continuamente de conectarse a algún Phidget dependiendo de los parámetros asignados, incluso trata de restablecer la conexión si el Phidget fue desconectado. Esto no garantiza que el objeto *phid* que se ha creado se pueda utilizar correctamente, para ello se recomienda utilizar los eventos *AttachEvents* y *DeattachEvents* o verificar la propiedad *isAttached*. Pero para efectos prácticos en este ejemplo se utiliza el objeto *phid* directamente.

```
phid.open(“localhost”, 5001); //Habilita el objeto phid
```

Cuando se trabaja con Phidgets en la plataforma Flash se recomienda utilizar la programación orientada a eventos. La sintaxis usada para leer el valor de algún sensor del

objeto *PhidgetInterfaceKit* se expresa de la siguiente manera:

```
phid.addListener(PhidgetDataEvent.SENSOR_CHANGE, onSensorChange);  
function onSensorChange(evt:PhidgetDataEvent){  
    trace (evt.Data); //Imprime el valor en consola  
}
```

Una vez que se tiene el código escrito en el fotograma se guarda el proyecto y se corre la película Flash. La función *onSensorChange* se ejecutará cada vez que ocurra un cambio en cualquiera de las entradas analógicas del *PhidgetInterfaceKit* y muestra un valor en consola de 0 a 1000. Algunos valores se pueden leer y escribir directamente sin la necesidad de estar a la espera de cambios en el sensor y usarlos como alternativas para la programación orientada a objetos. Para obtener el valor de un sensor se utiliza la función *getSensorValue(index:int)* donde *index* es la posición de la entrada analógica en la que se encuentra conectado el dispositivo, además la función *setOutputState(index:int, val:boolean)* se utiliza para proporcionar el valor lógico uno u cero a la salida digital en la posición *index*.

Capítulo 4

Evaluación del uso de Phidgets

- Describir cómo se hicieron las encuestas.
- Las primeras encuestas que fueron inventadas hay que describir como se desarrollaron. (Algo también inventado).
- Describir las dimensiones, factores e indicadores que permitieron realizar las encuestas.
- Demostrar que las dos primeras encuestas se hizo unas pruebas para demostrar que funcionan para lo que fueron hechas.
- Dar algunos resultados de las primeras encuestas. Por ejemplo, que no se recomienda utilizar los dispositivos Phidgets con alumnos de tronco común porque demostraron no tener interés ya que algunos van a carreras que no están relacionadas con las de electrónica o computación.

El objetivo de este capítulo es mostrar como se elaboraron las herramientas de evaluación para conocer el interés y la aceptación del uso de dispositivos Phidgets en un ambiente universitario. Gracias al análisis e interpretación de resultados se presentan gráficas de los datos que permiten mostrar mejor y con mayor eficacia los resultados.

4.1 Elaboración de herramienta para la evaluación del uso de Phidgets

Se elaboraron tres diferentes herramientas de evaluación en forma de encuestas personales de satisfacción. Dichas encuestas están dirigidas a estudiantes de las carreras de electrónica, computación y mecatrónica de la Facultad de Ingeniería de la Universidad Autónoma de Baja California (UABC).

Para definir las preguntas que sirvieron para recopilar información, se determinó primeramente que las dimensiones que se querían medir son Uso y Desarrollo

4.2 Evaluación a estudiantes

Referencias

Molina, M., “Intervenir en ciencia para el desarrollo”,
<http://educacionadebate.org/2011/04/04/8192/>, 2011.

Educación Hoy, “¿La educación de ciencias, en peligro?”, Boletín UNESCO N° 11,
<http://www.unesdoc.unesco.org/images/0013/001368/136850s.pdf>, ISSN: 1814-3989,
Octubre - Diciembre, 2004.

Perazo, C., “La Argentina, con déficit de ingenieros”,
<http://www.lanacion.com.ar/1293995-la-argentina-con-deficit-de-ingenieros>, 2010.

Berriolope, D., “Robótica: semillero de ingenieros”,
<http://www.informationweek.com.mx/columnas/robotica-semillero-de-ingenieros/>, Abril 2,
2012.

Greenberg, S., Boyle, M., Customizable Physical Interfaces for Interacting with
Conventional Applications, 2002.

Greenberg, S., Fitchett, C., Phidgets: Easy development of physical interfaces through
physical widgets, 2001.

Paret, Dominique, RFID at ultra and super high frequencies: theory and application, 2005.

Chen, K.Y., Bassett, D.R., Jr., The technology of accelerometry-based activity monitors:
Current and future, 2005.

Hegde, Ganesh S., Mechatronics, 2010

Giacomin, J., What is Human Centred Design?, 2012

International Standard, Ergonomics of Human-System Interaction - Part 210: Human-
Centred Design for Interactive Systems, 2010.

Wigod, D. and Wixon, D., Brave NUI World: Designing Natural User Interface for Touch
and Gesture, 2011

Wu, T.F., Chen, M.C. and Wu, C.F., The Performance of Mouse Proficiency for Adolescents with Intellectual Disabilities, 2010.

Girard, S. and Johnson, H., Educational Software for Children: Analysis of Interaction Techniques for Direct Manipulation, 2009.

Srikulwong, M., Multimodal Interaction in a Ubiquitous Environment, 2007.

Horan, P., Using Rich Pictures in Information Systems Teaching, 2000.