Instituto Superior Politécnico de Córdoba

Marco: Practicas del periodo #2

Objetivo: Creación de proyectos

Descripción

En el marco del aprendizaje basado en proyectos, del periodo #2: "Creación de proyectos", se propone al alumno el desarrollo de un dispositivo pre-thing.

La modelización del dispositivo constituye un marco de referencia para la organización de las actividades que involucran todas las etapas del desarrollo. La representación del ciclo de vida define los estados por los que pasa un producto (Software, Hardware) y su implementación (Simulación - Prototipado) define el conjunto de actividades no ordenadas en el tiempo que requiere su desarrollo.

Para ello surgen los marcos de trabajo, cuyo objetivo final se resumen en la obtención de productos perpetuos en el tiempo a base de la evolución de estos, sostenidos por un equipo adaptado.

Introducción

Para el desarrollo de un dispositivo, antes de empezar propiamente con su desarrollo e implementación, se debe resolver la organización del equipo y de las herramientas a utilizar.

El equipo ideal

El equipo ideal no existe, así como tampoco el producto ideal. Siempre se va a necesitar mejorar los aspectos de ambos en una adaptación constante, en pos de la mejora del producto. Si la definición habitual de un equipo de alto desempeño implica el sostenimiento de la excelencia en el tiempo; en el desarrollo de dispositivos se debe aspirar al alto rendimiento y estándar del equipo en base al alto performance del producto o dispositivo.

Por esto, es que, más allá del marco o metodología ágil que elija para su implementación es sumamente necesario ser preciso en la determinación de los requerimientos. Esto significara sin duda, el primer escalón para el éxito o el fracaso. Una vez asegurado esto, el mantenimiento evolutivo del producto hará el resto.

Las Herramientas

A diferencia de los equipos, las herramientas ideales si existen. Para el análisis, diseño, implementación y pruebas debe asegurarse de contar con las mejores herramientas software y hardware. Son muchas las referencias que existen al respecto. En esta materia vera muchas de ellas, pero al final su propia experiencia será la guía correcta.

El ciclo de vida de un Dispositivo

Como se ve en la figura abajo el mismo consta de:

i) **El requerimiento:** El mismo debe ser detallado en un manual de forma precisa, referida a su función u operación. Esto debe ser ejecutado al detalle.

ii) **Análisis:** Acá se define el producto en datos, se establece la solución al requerimiento y de ella se infiere: materiales, tiempo de desarrollo y costo.

iii) **Diseño:** Del análisis se propone el diseño mas optimo (costo - beneficio).

iV) **Implementación:** Prototipado - Codificación

V) **Prueba:** El prototipo debe ser 100% funcional.

Vi) **Validación:** Se prueba el dispositivo en campo.

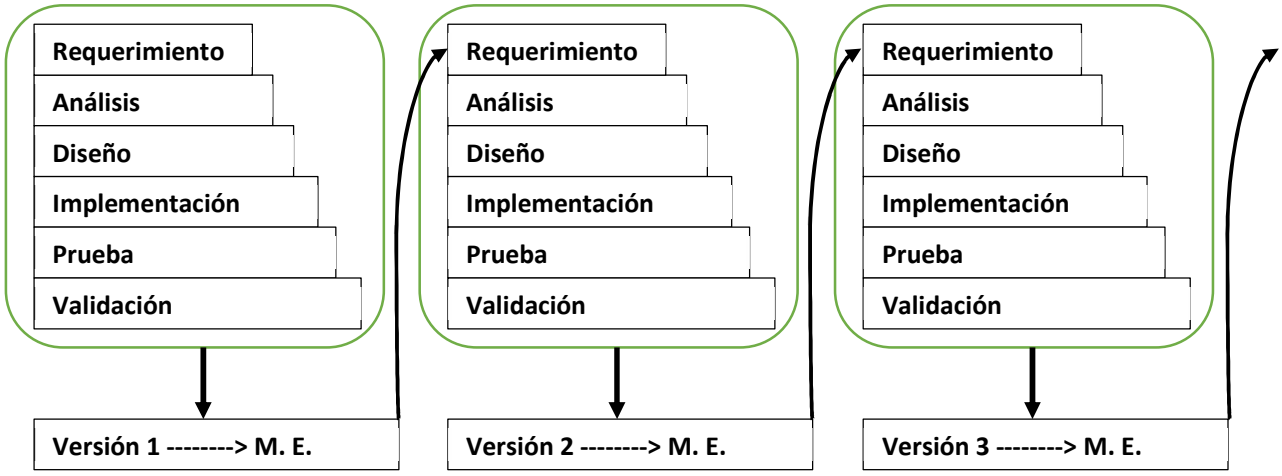


Figura 1.1 Ciclo de vida de un Dispositivo

Ejemplo #1 Crear una alarma de hogar.

i) Requerimiento

El requerimiento consta de 2 partes, 1 del lado del cliente, llamado de uso. Y otro generado por el equipo de diseño, el requerimiento de desarrollo.

Requerimiento de Uso:

Alarma para el hogar, que me permita proteger al mismo de la intrusión de ladrones. Y avise a las personas en caso de robo.

Requerimiento de desarrollo:

Sistema de Comunicación WIFI + GSM + Línea. Lo que permite monitorear, o recibir alarmas por varios medios cuando un detector se active. 99 zonas para sensores inalámbricos (comunicación con la central por un sistema cerrado y encriptado de Radio Frecuencia) y 4 para cableados. Admite cámara ip.

Además de sonar la Sirena frente a una intrusión puede elegir que le avise a su celular de 3 maneras distintas (puede utilizar 1 sola o las 3 juntas) WIF

1- WIFI: Compatible con la aplicación TUYA (la más popular en Domótica)
2- GSM: le coloca a la central un chip de cualquier compañía admite chip 2G - 3G - 4G- 5G y podrá comunicarse , recibir alertas y configurar la misma desde cualquier celular.
3- LINEA: Si tiene una línea fija telefónica en el sitio donde coloque la central y pueda conectarla como cualquier teléfono, desde ahí la central utilizará dicha línea para comunicarse con los teléfonos o celulares que desee

Tiene una variedad muy amplia de funciones que puede manejar desde su celular, como el encendido, apagado, modo hogar (apagar ciertos sensores mientras el resto queda encendido), pánico del equipo, modo inteligente.

Registro de todos los eventos, manejar varios dispositivos distintos desde la misma aplicación con la posibilidad de utilizar un relé incorporado a la central para incorporar cualquier dispositivo 12v. Incluye dentro de la sirena exterior una batería autorecargable 12v-1.3A para casos de corte de luz que permite una autonomía todo el sistema de 10 horas aproximadamente

Sistema de fácil instalación sin cables entre la central y los sensores. La central se conecta a corriente y los sensores llevan pilas, la distancia de cobertura desde la central a los sensores/controles remotos es de unos 30 metros en lugares donde hay paredes de por medio y unos 60 metros en lugares abiertos.

COMPONENTES

A. CENTRAL DE ALARMA con comunicacion vía WIFI -GSM- LINEA. Batería incorporada autorecargable . Panel con teclado y Display de fácil lectura, voz en español.

B. SENSOR DE MOVIMIENTO INALÁMBRICO.(2 unidades). Nano tecnología (cabe dentro de una mano) Para espacios interiores cubre entre 6-8 metros de distancia. Tecnología de avanzada dual para evitar falsas alarmas compacto. Funciona con pila (incluida) 1 pila CR2450 con una durabilidad de 12 meses aproximadamente

C. SENSOR MAGNÉTICO DE APERTURA PUERTA/VENTANA INALÁMBRICO (2 unidades) dos cuerpos que al separarse emiten la señal de alerta a la central. Indicador de batería baja. Funciona con 1 pila CR-2032 (incluida) con una durabilidad de 12 meses aprox.

D. CONTROLES REMOTOS (2 Unidades) Permite el comando a distancia de la central posee 4 teclas: encendido general- apagado general- encendido parcial (Modo hogar) - pánico. Funciona con pila A-27 con una durabilidad de 12 meses aprox.

E. SIRENA CABLEADA con destellador para exterior muy potente doble tono 120 db con batería interior 12v 1.3a autorecargable que abastece a todo el sistema de alarma durante 10 hs aprox en caso de corte

F. SIRENA CABLEADA apta para interior. La misma se conecta a la central, de tamaño pequeño para su fácil y discreta ubicación y potente en espacios interiores 105 db.

G. FUENTE DE ALIMENTACIÓN 12-220v conectada a la central

H. CARTEL DE PROPIEDAD PROTEGIDA, de material plástico

I. MANUAL en español

CARACTERÍSTICAS

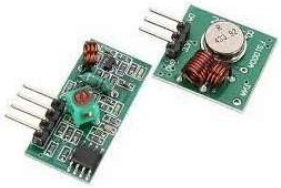
• Fácil instalación. No se necesitan conocimientos previos
• Aplicación para celular (APP) en español, amigable e intuitiva
• Se puede comandar y configurar desde cualquier celular solo entrando a la App de manejo o vía GSM.
• Se puede comandar desde cualquier teléfono por llamado de voz ante falta de cobertura wifi (6 números llamado de voz - 3 números SMS)
• Sensores de última generación con antena incorporada al sensor.
• Permite configurar armado y desarmado por tiempo
• Batería autorecargable 12v-1.3A para casos de corte de luz que permite una autonomía todo el sistema de 12 horas aproximadamente.
• Es muy fácil agregar sensores inalámbricos de todo tipo en cualquier momento (humo, gas, barreras, pánico, controles, rotura de vidrio) .
• Permite incorporar sensores cableados
• 99 Zonas inalámbricas + 4 Cableadas

ii) Análisis

*Consideraciones:

a) Si bien el ecosistema IoT Tuya no está proporcionado propiamente, se corresponde con una solución genérica de soporte.

b) Los sensores inalámbricos trabajan en el rango 433 mhz utilizando los protocolos pt2262 y ev1527. Se puede utilizar el shield RF.



Por lo que los sensores pueden ser genéricos, o bien propietarios manejando estos protocolos.

c) La cámara ip maneja una computación igual o más desarrollada que la central de alarma, la comunicación de esta con la central puede ampliarse a la de los transmisores rf(433 MHZ). Por otra parte, amerita un desarrollo propio, y aun así debe integrarse al soporte app para el celular, en este caso app tuya.

Por lo mencionado el desarrollo inicial será basado en la central de alarma, considerando la parte de la cámara ip solo como un sensor más. Pero se debe tener en cuenta el valor que le proporciona la cámara al sistema de alarma monitoreado, casi como un detector universal de contexto y de funcionamiento.

De esta forma la central consistiría en módulo de control, módulo de visualización y modulo teclado (con una tft, esto se integraría en un solo modulo), módulo de comunicación (GSM,

LAN, RF, Telefónico) Veamos como seria el diagrama en bloque de la implementación.

Finalmente, está claro, que el desarrollo del dispositivo tiene una finalidad comercial, por lo que debe maximizar beneficios para todos los canales: Desarrollador, Fabricante, Vendedor y Cliente. Siendo este ultimo el mas importante.

Pero veamos el punto de vista del desarrollador, necesito para mi desarrollo lo siguiente:

a) Realizar el prototipo final lo mas rápido posible.
b) Lo mas barato posible.
c) Respetando los requisitos y la estética del producto.

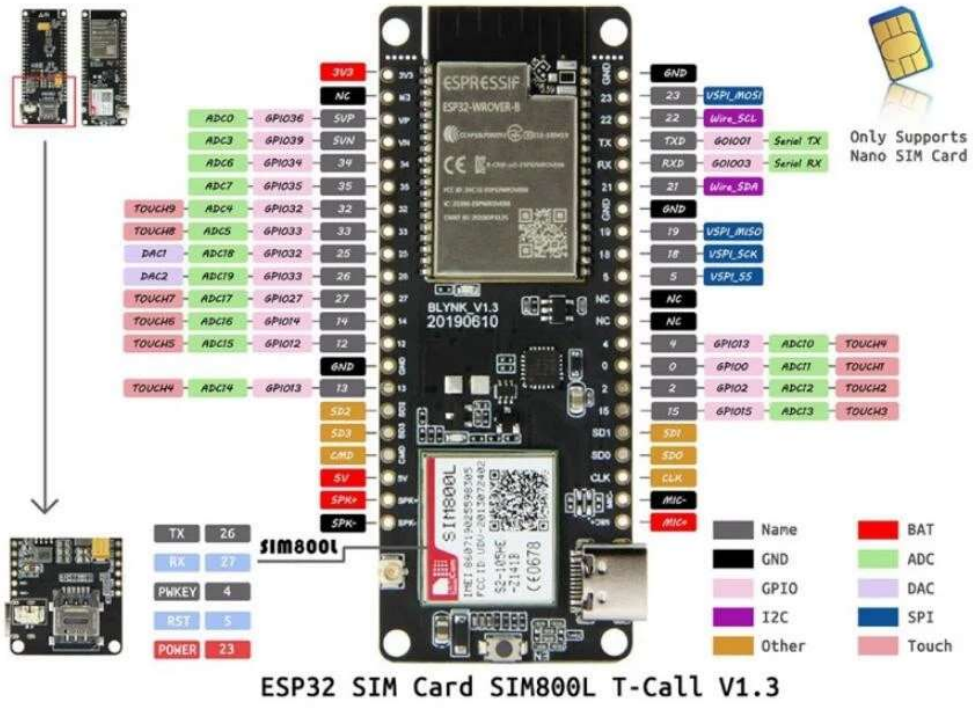
Con respecto al ultimo punto, los requisitos se deben implementar desde un principio. Olga decir, que el manual del mismo debe ser impoluto. En cuanto a la estética del producto, las consideraciones iniciales pueden ser con respecto a los efectos sobre el tamaño y disposición de los componentes electrónicos. Luego esta en el diseñador del producto la estética del mismo.

.iii) Diseño

De la misma forma el diseño del desarrollador es separado del diseño estético del producto, aunque a veces este ultimo condiciona al primero.

Con respecto al diseño de desarrollo, la solución plausible seria utilizar

Controlador: TTGO t-call V1.4 ESP32 módulo inalámbrico antena SIM tarjeta SIM módulo SIM800L y antena GSM/GPRS



Pantalla: **NX8048P070-011C-Y** Nextion 5” capacitiva.

Sensores: Control Remoto, Magnéticos, PIR @ 433 mhz. Protocolo ev1527. RFID RC522 @ 13,56 mhz. {Teclado de pantalla}

Actuadores: Sirena, Campana, Parlante.

iii) Implementación.

Una posible solución seria como máquina de estado, es decir, la alarma puede estar básicamente en 4 estados: Desactivada, Activada, Disparo, Disparo silencioso. Este último sería útil si no quiero que suene la sirena o la campana, pero que si se realice el pedido de S.O.S por mensaje o llamada. Analizaremos bien esto de los estados mas adelante.

Nos enfoquemos en los sensores, sus características y comportamiento.

El sensor magnético. Este sensor se utiliza generalmente para la detección de apertura de puertas y ventanas. La alarma

soportara 99 de este tipo de sensores. Los cuáles serán podrán ser identificados para saber dónde fue la intrusión, además de poder configurarse a la zona que pertenece.

El sensor PIR. Este sensor se utiliza para detectar presencia, se suele conectar en pasillos y espacios de la casa como el living, comedor, garaje, etc. Igual que los magnéticos la alarma soportara 99 de estos dispositivos, que también tendrán un id y se podrá configurar su zona.

Ahora bien, que es una zona y por que es necesaria. Lo zona es un concepto que se utiliza para variar el comportamiento de nuestros sensores de acuerdo con el estado de la alarma. Es decir, cuando yo activo la alarma con el teclado o con el RFID no quiero que la alarma se active inmediatamente, sino que tenga una temporización que me permita retirarme del lugar; y pasado este tiempo se active. De igual manera si quiero activar la alarma cuando estoy dentro de casa, no quiero que el sensor PIR del pasillo de los dormitorios, se active si me levanto por la noche para ir al baño, o si quiero ir a la cocina por un vaso de agua. El comportamiento de los sensores puede ser determinado según la zona a la que se asignaron. Las zonas son:

Zona parcial: Desactiva los sensores que pertenecen a este grupo cuando se activa la alarma en forma parcial.

Zona segura: Los sensores se comportan de forma ordinaria. Es decir, se disparan la alarma si se activan.

Zona temporizada: Se le da una temporización a la activación (teclado – rfid) o al disparo (magnético-pir). Se pueden configurar varias zonas temporizadas, pero existen 2 predefinidas. *Zona teclado* (2 minutos). *Zona servicio* (5 minutos).

Ya descripto el comportamiento de los sensores, nos enfoquemos en la alarma. La misma, puede estar en los siguientes estados (definidos por nosotros los desarrolladores):

Desactivado: En este estado la alarma indica que los sensores magnéticos y PIR no disparan la alarma. Aun así, el teclado, el control remoto o el llavero RFID si podrían Disparar la alarma, y también por supuesto Activarla.

Activado: En este estado la alarma indica que los sensores magnéticos y PIR cuando detectan disparan la alarma. Se puede volver a desactivar, con el teclado, el control remoto o un RFID.

Disparo: En este estado, suena la sirena, la campana y el discador llama y envía mensaje si estuviera configurado.

Disparo Silencioso: No suena la sirena ni la campana. Pero el discador ejecuta la llamada de S.O.S y se envían los mensajes configurados.

De esta forma la palabra de estado que determinara el funcionamiento de la alarma podría ser:

— — — — —

E T Z ID V

E: Determina el estado de la alarma (1 digito)

T: Determina el tipo del sensor (1 digito)

Z: Determina la zona del sensor (2 dígitos)

ID: El identificador del sensor (2 dígitos)

V: El valor actual del sensor. (1 digito)

A continuación, se desarrolla la función de control.

```
#include <arduino.h>
// #include "alarma.h"


/* Autores */
// Alumnos ISPC:
/*                 */
/******************/

// Descripcion
// Crear una alarma para casa, como una maquina de estado.
// La alarma se va a comportar de acuerdo a una palabra de estado
// que estara constituida de la siguiente manera:
//
//   Estado: La alarma puede estar activada, desactivada, disparo,
disparo silencioso o configuracion. 1 digito, de 0 a 9.
//   Tipo: Indica el tipo de sensor: magnetico, PIR, control, llavero,
rfid, teclado.  1 digito, de 0 a 9.
//   zona: Indica la zona a la que pertenece el sensor. Esta puede ser
zona segura, zona temporizada, zona parcial.
//         2 digitos, de 00 a 99.
//   Id: Indica el id del sensor. 2 digitos, de 00 a 99.
//   valor: Indica el valor del sensor. activado, desactivado. 0 o 1.


// includes
// definiciones: globales, constantes, clases, macros, funciones,
variables

// Implementar la clase sensor.
class sensor {
  public:
    sensor(int tipo, int zona, int id, int valor);
    int getTipo();
    int getZona();
    int getId();
    int getValor();
    void setTipo(int tipo);
    void setZona(int zona);
    void setId(int id);
    void setValor(int valor);
  private:
    int tipo;
    int zona;
    int id;
    int valor;
};

int sensor::getTipo() {
```

```
  return tipo;
}

int sensor::getZona() {
  return zona;
}

int sensor::getId() {
  return id;
}

int sensor::getValor() {
  return valor;
}

void sensor::setTipo(int tipo) {
  this->tipo = tipo;
}

void sensor::setZona(int zona) {
  this->zona = zona;
}

void sensor::setId(int id) {
  this->id = id;
}

void sensor::setValor(int valor) {
  this->valor = valor;
}




// Implementar la clase Alarma con sus estados: activada, desactivada,
disparo, disparo silencioso
// Ademas heredar de la clase sensor sus atributos y metodos.
class alarma : public sensor {
  public:
    alarma();
    void setEstado(int estado);
    int getEstado();
    void setTipo(int tipo);
    void setZona(int zona);
    void setId(int id);
    void setValor(int valor);
    long setPalabra(int estado, int tipo, int zona, int id, int valor);
  private:
```

```
   long palabraEstado;    // La palabra de estado esta formada por los
digitos de estado, tipo, zona, id y valor. En ese orden.
    int estado;
};

void alarma::setEstado(int estado) {
  this->estado = estado;
}

int alarma::getEstado() {
  return estado;
}

void alarma::setTipo(int tipo) {
  this->tipo = tipo;
}

void alarma::setZona(int zona) {
  this->zona = zona;
}

void alarma::setId(int id) {
  this->id = id;
}

void alarma::setValor(int valor) {
  this->valor = valor;
}

long alarma::setPalabra(int estado, int tipo, int zona, int id, int
valor) {
  palabraEstado = 0410000; // E= 0, T= 4, Z= 10, Id= 00, V= 0
  return palabraEstado;
}




// setup
void setup(){
    Serial.begin(9600);
    Serial.println("Alarma");
    Serial.println("Iniciando...");
    alarma.inicializar();
    Serial.println("Iniciado");
}

// loop
void loop(void){
```

```
// objeto alarma
    alarma alarma;
// obtener la palabra de estado, que debe tener 7 digitos: 1 digito de
estado, 1 digito de tipo,
//                                                      2 digitos de
zona, 2 digitos de id, 1 digito de valor.
    long palabraEstado = alarma.setPalabra(alarma.getEstado(),
alarma.getTipo(), alarma.getZona(), alarma.getId(), alarma.getValor());

// Inicio de maquina de estado de acuerdo al valor de la palabra de
estado
// modo de operacion:
//  0: 'DESACTIVADA'
//  1: 'ACTIVADA'
//  2: 'DISPARO'
//  3: 'DISPARO SILENCIOSO'
//  4: 'CONFIGURACION'
// Declarar una enumeracion para mostrar el estado de la alarma.
    enum estado {
        DESACTIVADA,
        ACTIVADA,
        DISPARO,
        DISPARO_SILENCIOSO,
        CONFIGURACION
    } estado;
    // estado = DESACTIVADA;

    switch (estado){
        case DESACTIVADA:
            Serial.println("Desactivada");
            break;
        case ACTIVADA:
            Serial.println("Activada");
            break;
        case DISPARO:
            Serial.println("Disparo");
            break;
        case DISPARO_SILENCIOSO:
            Serial.println("Disparo silencioso");
            break;
        case CONFIGURACION:
            Serial.println("Configuracion");
            break;
        default:
            Serial.println("Error");
            break;
    }
```

```
}
```

Bibliografia:

proyectos

¿Qué es un proyecto?

¿Qué es la gestión de proyectos?

Curso de Fundamentos de proyectos según PMI y el PM4R

Metodologías y Gestión de Proyectos agiles (atlassian)

Alarmas

Creación de alarma con arduino (protocolos PT2262 y EV1257)