

Desarrollo de un prototipo Arduino - Móvil
y Geoposicional en el área de la Telemedicina
para el monitoreo remoto de personas
diabéticas a través de la web con servicios
extendidos geo referenciados

MSc. Yair Enrique Rivera Julio

ISBN: 978-958-58187-8-1



CORPORACIÓN UNIVERSITARIA
AMERICANA®
INSTITUCIÓN UNIVERSITARIA

Rivera Julio, Yair Enrique.

Desarrollo de un prototipo Arduino-móvil y geoposicional en el área de la telemedicina para el monitoreo remoto de personas diabéticas a través de la web con servicios extendido geo referenciados / Yair Enrique Rivera Julio. Barranquilla : Corporación Universitaria Americana, 2014.

47 p. ; xx cm.

ISBN: 978-958-58187-8-1

1. Medicina – Aparatos e instrumentos. 2. Telemedicina. I. Tít.

616.462 R621d 2014 cd 21 ed.

Corporación Universitaria Americana-Sistema de Bibliotecas

Presidente
JAIME ENRIQUE MUÑOZ

Rectora Nacional
ALBA LUCÍA CORREDOR GÓMEZ

Rector Sede Medellín
ALBERT CORREDOR GÓMEZ

Vicerrector Académico
LÁSTER ALFONSO GUTIÉRREZ CUADRO

Director Centro de Investigaciones
JUAN CARLOS MIRANDA

Director de Publicaciones
CARLOS FEDERICO MIRANDA MEDINA.

Desarrollo de un prototipo Arduino móvil y geo posicional en el área de la telemedicina para el monitoreo remoto de personas diabéticas a través de la WEB con servicios extendidos georeferenciados.

Por MSc. Yair Enrique Rivera Julio ©

No. 2 - 2014

ISBN: 978-958-58187-8-1

Todos los derechos reservados. Ninguna parte de esta publicación puede ser reproducida, almacenada en sistema recuperable o transmitida en ninguna forma o por medios electrónico, mecánico, fotocopia, grabación, u otra, sin la previa autorización por escrito de Sello Editorial Coruniamericana y del autor. Los conceptos expresados en este documento son responsabilidad exclusiva del autor y no necesariamente corresponden con los de la Corporación Universitaria Americana y da cumplimiento al Depósito Legal según lo establecido en la Ley 44 de 1993, los Decretos 460 del 16 de Marzo de 1995, el 2150 de 1995, el 358 de 2000 y la Ley 1379 de 2010.

Sello Editorial Coruniamericana
Calle 72 No. 41C-64
selloeditorialcoruniamericana@coruniamericana.edu.co

**DESARROLLO DE UN PROTOTIPO ARDUINO-MÓVIL Y GEO POSICIONAL EN EL
ÁREA DE LA TELEMEDICINA PARA EL MONITOREO REMOTO DE PERSONAS
DIABÉTICAS A TRAVÉS DE LA WEB CON SERVICIOS EXTENDIDOS GEO
REFERENCIADOS.**

GRUPO DE INVESTIGACIÓN

AGLAIA COL0081361

EJECUTORES

COORDINADOR DEL PROYECTO: MSC. YAIR ENRIQUE RIVERA JULIO

**TRABAJO DE INFORME FINAL DE INVESTIGACIÓN
DIRIGIDO AL DEPARTAMENTO DE INVESTIGACIONES**

**CORPORACIÓN UNIVERSITARIA AMERICANA
BARRANQUILLA, COLOMBIA**

2014

Contenido

Introducción	2
Objetivo General	4
Objetivos Específicos	4
Planteamiento de problema	5
Justificación y estado del Arte	5
Metodología	8
Diseño y modelado	8
Desarrollo de la arquitectura	9
Arquitectura de hardware	9
Codificación del sistema	11
Recepción de los Datos	18
Arquitectura de servicios espaciales	22
Capa de aplicación	22
Capa de servicios	23
Capa de acceso a datos	23
Flujo grama de servicios	23
Implementación y almacenamiento	28
Base de datos geográfica y visualización	31
Evaluación del Sistema	33
Criterios de selección de la tecnología	36
Resultados	39
Conclusión	42
Referencias	44

Resumen

La iniciativa del proyecto pretende diseñar la arquitectura de un Sistema de información geográfica ligado a un prototipo electrónico móvil basado en módulos Arduino, los cuales tiene sensores integrados para telemedicina en el monitoreo de personas diabéticas, donde se definen los parámetros necesarios para la digitalización y consolidación de la información en un sistema persistente, esto con el fin de brindar servicios extendidos y complementarios por un sistema básico de localización con base a los datos actualizados. La comunicación del sistema se da a través de la red de paquetes GPRS de la red celular GSM, lo que permite mantener al paciente diabético en una movilidad y una conectividad constante, con la red y los sistemas de información configurada.

Utilizando tecnologías de generación, como una base de datos geográfica en Mysql y aplicaciones que corren en la Web, se pueden ofrecen servicios de geo posicionamiento como servidores de mapas al enviar desde el dispositivo móvil los niveles censados de glucosa junto con las coordenadas de geo posicionamiento, latitud y longitud por medio del protocolo de comunicación HTTP, el cual permite enlazar el dispositivo móvil hacia el sistema persistente.

Palabras clave: Arduino, GSM, HTTP,GRPS.

Introducción

Siguiendo a Ibáñez, (2011), Meulen y Pettey, C. (2012)., durante los últimos años han venido creciendo las tecnologías emergentes, especialmente en tecnología móvil y telecomunicaciones, los cuales permiten mantener al usuario cada día mas conectado a la web, lo que de alguna manera ha incentivado el desarrollo de sistemas de información en cualquier área del desarrollo humano, especialmente en telemedicina móvil, aunque esta área no se ha explotado en todo su potencial, ya que son pocas las soluciones “open source”, de telemedicina móvil donde complementa sus servicios extendidos en la geo referenciación, es decir tratar de monitorear cualquier señal del paciente en tiempo real. Con base a Mahmoud (2010), el crecimiento de esta tecnología emergente estaría graduado en la medida en que se ofrezcan servicios que realmente cubran las necesidades de los usuarios, una condición necesaria para que esto se dé es la movilidad pero faltan servicios complementarios para que el crecimiento se mantengan, de aquí que la convergencia de las comunicaciones, los sistemas de información, movilidad y los servicios médicos, la congruencia de estos servicios podrían ofrecer servicios más inteligentes, funcionales e interactivos, los cuales faciliten el intercambio inmediato de los datos, de información entre centros de atención y sitios o pacientes ubicados en zonas remotas o de difícil acceso, esto permitirá mejorar los servicios o una asistencia médica algo limitada, específicamente aquellas establecidas por los recursos médicos y especialistas de alto nivel.

El desarrollo de esta arquitectura telemática permitirá una interacción entre los dispositivos móviles en telemedicina de forma directa, posibilitando a través de la red de datos de la internet, la adquisición de señales biológicas, en especial sobre el nivel de glucosa en la sangre, de

relevancia para aquellos pacientes diabéticos, y almacenando sus valores en sistemas itinerantes para su posterior consulta, desde cualquier dispositivo conectado a la web.

Objetivo General

Diseñar un sistema de información Geográfico sobre la web para telemedicina móvil basada en módulos Arduinos, los cuales se conectan a sensores que miden el nivel de glucosa en la sangre y que través de un *Middleware* permitirán comunicarse entre otros dispositivos inteligentes para el intercambio de información en tiempo real.

Objetivos Específicos

- Programar una placa controladora junto a unos sensores especializados que permitan determinar el nivel de glucosa en la sangre y el envío de información a través de la red de datos *GPRS* de la red celular.
- Realizar una codificación del sistema para el envío de la información vía *HTTP*.
- Implementar un sistema básico de localización que permita a través de radionavegación, manejar la ubicuidad y tracking del paciente diabético.

Planteamiento de problema

¿Cómo generar servicios extendidos basados en geo localización para telemedicina móvil y ubicua basados en módulos Arduinos para pacientes diabéticos ubicados en zonas remotas y de difícil acceso?

Justificación y estado del Arte

Dentro del marco regional colombiano, el aumento de la diabetes en personas que se encuentran ubicadas en zonas alejadas y de difícil acceso originando también la atención médica especializada dentro de estas mismas comunidades, sin embargo debido a la escasez médica especializada en diferentes regiones, los pocos concentrados principalmente en las diferentes ciudades, por esta razón deben desplazarse desde su lugar de trabajo hasta las zonas más cercanas de atención especializada, en otras ocasiones la persona afectada es la que se desplaza al centro especializado. Ahora bien según López y Blobel(2007), Mykkänen,Korpela, Ripatti, Rannanheimo y Sorri, J. (2007), si examinamos el incremento de la accesibilidad de las comunicaciones de acceso junto a servicios convergentes, estas podrían servir de apoyo a los recursos médicos especializados en estas zonas rurales y de difícil acceso, servicios que según el Programa (PROMESA-2012), serían de manera principal más que indispensables, para el desarrollo, y operación de infraestructuras telemáticas inclusivas y colaborativas que apoyarían al sector de la salud, nuevas políticas de operación que diera respuestas a las diferentes necesidades y demandas de la región, especialmente al tratamiento remoto de personas con diabetes, las cuales de alguna u otra manera le mejoraría las condiciones de vida, esto según Pérez, Martínez, Bustamante, Betancur y Torres(2011), a lo anterior le podemos abonar los últimos avances a nivel científico:

Lightweight and Confidential Data Discovery and Dissemination for Wireless Body Area Networks, Daojing He ; Chan, S. ; Yan Zhang ; Haomiao Yang Biomedical and Health

Informatics, IEEE Journal of Volume: 18 , Issue: 2

Digital Object Identifier: 10.1109/JBHI.2013.2293620

Publication Year: 2014 , Page(s): 440 - 448

IEEE journals & magazines

A Health-IoT Platform Based on the Integration of Intelligent Packaging, Unobtrusive Bio-Sensor and Intelligent Medicine Box

Yang, G. ; Xie, L. ; Mantysalo, M. ; Zhou, X. ; Pang, Z. ; Xu, L. ; Kao-Walter, S. ; Chen, Q. ; Zheng, L.

Industrial Informatics, IEEE Transactions on

Volume: PP , Issue: 99

Digital Object Identifier: 10.1109/TII.2014.2307795

Publication Year: 2014 , Page(s): 1

IEEE early access articles

WE-CARE: An Intelligent Mobile Telecardiology System to Enable mHealth Applications Anpeng Huang ; Chao Chen ; Kaigui Bian ; Xiaohui Duan ; Min Chen ; Hongqiao Gao ; Chao Meng ; Qian Zheng ; Yingrui Zhang ; Bingli Jiao ; Linzhen Xie

Biomedical and Health Informatics, IEEE Journal of

Volume: 18 , Issue: 2

Digital Object Identifier: 10.1109/JBHL.2013.2279136

Publication Year: 2014 , Page(s): 693 - 702

Designing High Performance Web-Based Computing Services to Promote Telemedicine
Database Management System

Hababeh, I. ; Khalil, I. ; Khreishah, A.

Services Computing, IEEE Transactions on

Volume: PP, Issue: 99

Digital Object Identifier: 10.1109/TSC.2014.2300499

Publication Year: 2014 , Page(s): 1

Smart Embedded Medical Diagnosis using Beaglebone Black and Arduino.

Ch Srikanth, DS Pradeep M,Sreeram Charan K.

Department of electronics and Communication, SLC institute of Engineering and
Technology, JNTU Hyderabad

International Journal of Engineering Trends and Technology(IJET)-Volume 8 Number 1-
feb 2014

A Ubiquitous Bases System for Health Care Monitoring

S. Monicka, C. Suganya, S. Nithya Bharathi, A.P.Sindhu

Velammal College of Engineering & Technology, Madurai, Tamilnadu, India

International Journal of Engineering Trends and Technology(IJSRET),ISSN 2278-0882 -
Volume 3 Issue 4,july 2014

Estas investigaciones han aportado elementos tecnológicos fundamentales para el desarrollo de este proyecto, el cual hará un énfasis en los servicios complementarios brindados con base a un sistema geográfico de información, donde se consolida toda la información enviada desde el dispositivo móvil biomédico, acorde a Pardo y Pardo (2011,p59-65).

Metodología

Diseño y modelado. Según Grover(2012), la arquitectura del sistema de los datos está representada a través de un diagrama UML (*Unified Modeling Language*), el cual se centra exclusivamente en la lógica del área especializada. Ver siguiente figura.

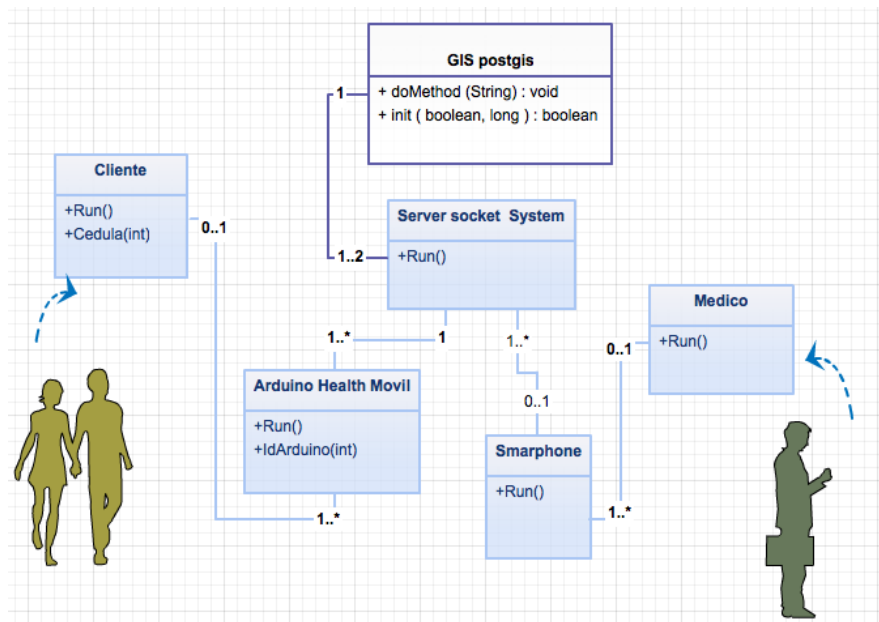


Figura 1. Modelado del sistema entre usuarios y clientes (Msc Yair Rivera,2014).

Desarrollo de la arquitectura. Está integrada por una plataforma *SIG* (“Sistema de información geográfica”), el cual obtiene toda una información cartográfica a través de los *API*’s de Google MAP, para junto con los datos de geo posicionamiento del dispositivo móvil biomédico, el cual envía sus coordenadas y datos censados a través de la web, el sistema monitorea en tiempo real el nivel de glucosa en la sangre de cada individuo, teniendo en cuenta los conceptos de Miller(2011), Daojing, Chan, Yan y Haomiao(2014).

La red de datos está proporcionada el sistema *GRPS* de la red *GSM* celular, donde el traspaso de la información se hace a través del protocolo http, y la recepción de los datos se da a través del script *PHP* que corre en el servidor Apache del *SIG*, según Pardo y Pardo (2011,p 58-92)

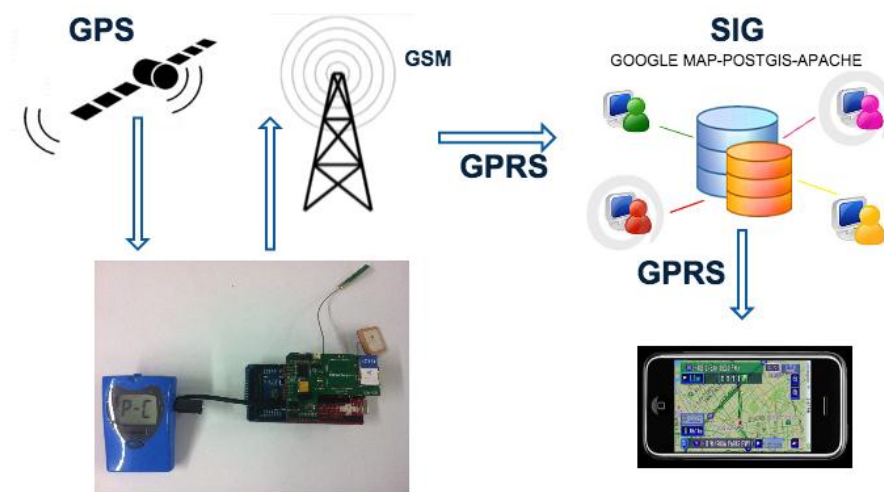


Figura 2. Modelado del sistema entre usuarios y clientes (Msc Yair Rivera,2014).

Arquitectura de hardware. Con base a Monk (2014, P 45-89), para el desarrollo de este prototipo biométrico se implementó en un hardware de código abierto o tipo *open source*, el *Arduino MEGA 250*, Ver Figura 3, un módulo de telemedicina para la toma de señales digitales del glucómetro *e-Health Sensor Shield* Ver Figura 4, un módulo de comunicación *GPS Y GSM*

para obtener las coordenadas de posicionamiento global y un módulo *GSM*, la cual trabaja con el sim908, con conexión a la red de datos *GPRS* de la red celular, indispensable para el envío de información hacia el *SIG* y posterior almacenamiento en la base de datos geográfica, ver figura 5.

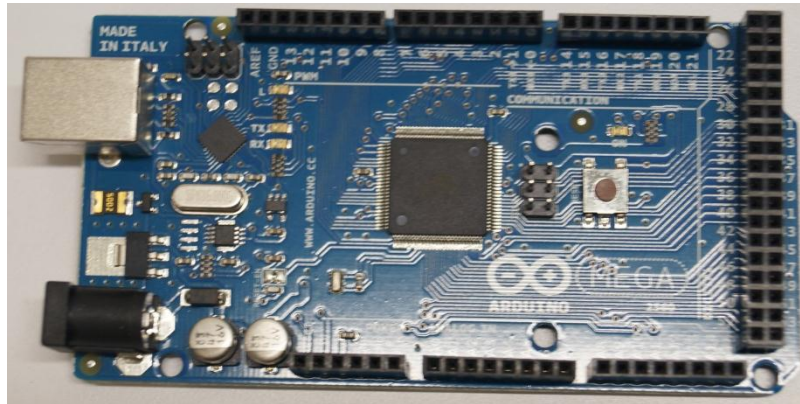


Figura 3. Placa base Arduino MEGA 250 (Msc Yair Rivera,2014).

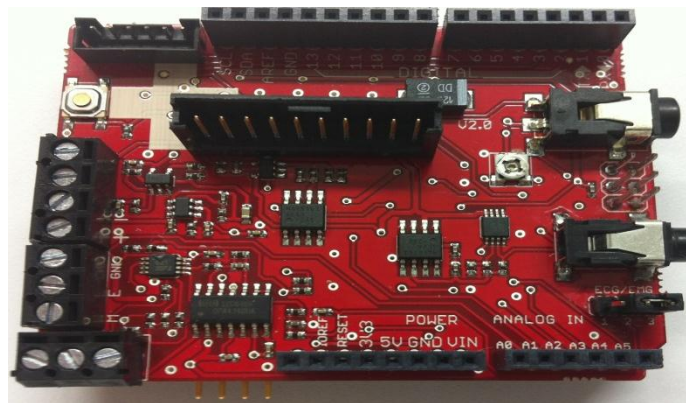


Figura 4. Placa Interface Arduino-Glucometer (Msc Yair Rivera,2014).

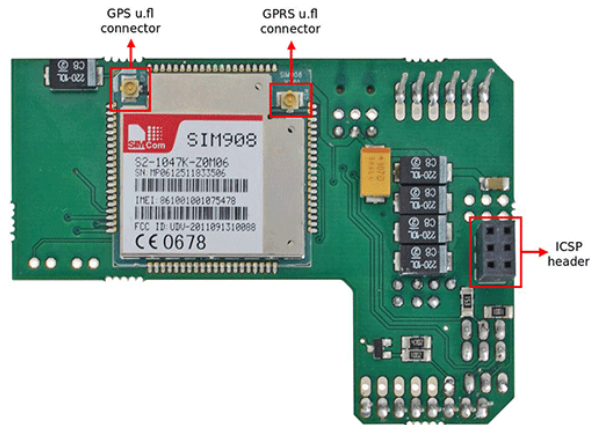


Figura5. Interconexión con la red de datos celular y la red satelital

GPS (<http://www.cooking-hacks.com/>, 2014).

Finalmente el dispositivo biomédico es el siguiente:

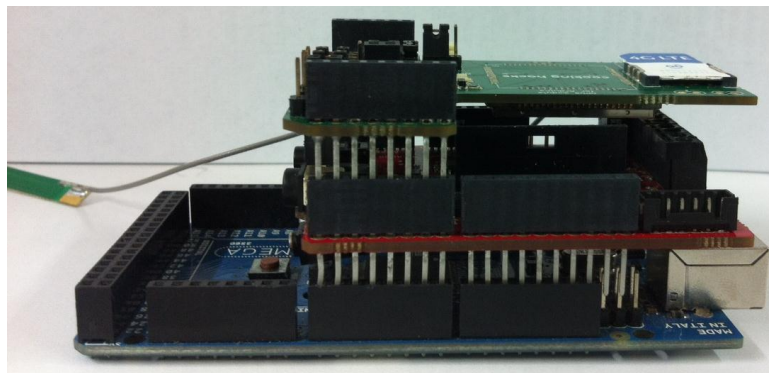


Figura 6. Prototipo Armado, módulo GSM y módulo HEALTH, vista frontal (Msc Yair Rivera, 2014).

Codificación del sistema. La siguiente instrucción de código muestra cómo trabaja el dispositivo, desde que lee los datos del glucómetro digital, hasta que hace la sincronización *GPS* y envía las coordenadas de geo posicionamiento a través de la red de datos *GPRS* en el sistema *GSM*.


```

#include <eHealth.h>
int8_t answer;
int onModulePin= 2;
char data[100];
int data_size;
char aux_str[30];
char aux;
int x = 0;
char N_S,W_E;

char url[] = "yair.hostingnubedigital.com/telemedicina/";
char frame[200];

char latitude[15];
char longitude[15];
char altitude[6];
char date[16];
char time[7];
char satellites[3];
char speedOTG[10];
char course[10];
String temps;

void setup(){

  eHealth.readGlucometer();
  Serial.begin(115200);
  delay(100);
  temps = glucosa();
  Serial.println(temps);
  pinMode(onModulePin, OUTPUT);
  Serial.begin(115200);
  Serial.println("Starting...");
  power_on();
  delay(3000);
  // sets the PIN code
  sendATcommand("AT+CPIN=****", "OK", 2000);
  delay(3000);
  // starts the GPS and waits for signal
  while ( start_GPS() == 0);
  while (sendATcommand("AT+CREG?", "+CREG: 0,1", 2000) == 0);
  // sets APN , user name and password
  sendATcommand("AT+SAPBR=3,1,\"CONTTYPE\",\"GPRS\"", "OK", 2000);
  sendATcommand("AT+SAPBR=3,1,\"APN\",\"web.colombiamovil.com\"", "OK", 2000);
  sendATcommand("AT+SAPBR=3,1,\"USER\",\"\", \"OK\", 2000);
  sendATcommand("AT+SAPBR=3,1,\"PWD\",\"\", \"OK\", 2000);
  // gets the GPRS bearer
  while (sendATcommand("AT+SAPBR=1,1", "OK", 20000) == 0)
  {
    delay(5000);
  }
}

void loop(){
  Serial.println("Starting GPS");
  // gets GPS data
  get_GPS();
  Serial.println("FINISH GPS");
  // sends GPS data to the scrip
  send_HTTP(temps);
}

```

```

    delay(5000);
}
void power_on(){
    uint8_t answer=0;
    // checks if the module is started
    answer = sendATcommand("AT", "OK", 2000);
    if (answer == 0)
    {
        // power on pulse
        digitalWrite(onModulePin,HIGH);
        delay(3000);
        digitalWrite(onModulePin,LOW);
        // waits for an answer from the module
        while(answer == 0){
            // Send AT every two seconds and wait for the answer
            answer = sendATcommand("AT", "OK", 2000);
        }
    }
}

int8_t start_GPS(){

    unsigned long previous;
    previous = millis();
    // starts the GPS
    sendATcommand("AT+CGPSPWR=1", "OK", 2000);
    sendATcommand("AT+CGPSRST=0", "OK", 2000);
    // waits for fix GPS
    while(( sendATcommand("AT+CGPSSTATUS?", "2D Fix", 5000) ||
        sendATcommand("AT+CGPSSTATUS?", "3D Fix", 5000)) == 0 ) &&
        ((millis() - previous) < 90000));
    if ((millis() - previous) < 90000)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}

int8_t get_GPS(){
    int8_t counter, answer;
    long previous;
    // First get the NMEA string
    // Clean the input buffer
    while( Serial.available() > 0) Serial.read();
    // request Basic string
    sendATcommand("AT+CGPSINF=0", "AT+CGPSINF=0\r\n\r\n", 2000);
    counter = 0;
    answer = 0;
    memset(frame, '\0', 100); // Initialize the string
    previous = millis();
    // this loop waits for the NMEA string
    do{

        if(Serial.available() != 0){
            frame[counter] = Serial.read();
            counter++;
            // check if the desired answer is in the response of the module
            if (strstr(frame, "OK") != NULL)
            {

```

```

        answer = 1;
    }
}
// Waits for the answer with time out
}
while((answer == 0) && ((millis() - previous) < 2000));

frame[counter-3] = '\0';

// Parses the string
strtok(frame, ",");
strcpy(longitude, strtok(NULL, ",")); // Gets longitude
strcpy(latitude, strtok(NULL, ",")); // Gets latitude
strcpy(altitude, strtok(NULL, ",")); // Gets altitude
strtok(NULL, ",");
strcpy(date, strtok(NULL, ",")); // Gets date
strtok(NULL, ",");
strtok(NULL, ",");
strcpy(satellites, strtok(NULL, ",")); // Gets satellites
strcpy(speedOTG, strtok(NULL, ",")); // Gets speed over ground. Unit is knots.
strcpy(course, strtok(NULL, "\r")); // Gets course
convert2Degrees(latitude);
convert2Degrees(longitude);
return answer;
}

/* convert2Degrees ( input ) - performs the conversion from input
* parameters in DD°MM.mmm' notation to DD.ddddd° notation.
*
* Sign '+' is set for positive latitudes/longitudes (North, East)
* Sign '-' is set for negative latitudes/longitudes (South, West)
*
*/
int8_t convert2Degrees(char* input){
    float deg;
    float minutes;
    boolean neg = false;
    //auxiliar variable
    char aux[10];
    if (input[0] == '-')
    {
        neg = true;
        strcpy(aux, strtok(input+1, "."));
    }
    else
    {
        strcpy(aux, strtok(input, "."));
    }

    // convert string to integer and add it to final float variable
    deg = atof(aux);

    strcpy(aux, strtok(NULL, "\0"));
    minutes = atof(aux);
    minutes /= 1000000;
    if (deg < 100)
    {
        minutes += deg;
        deg = 0;
    }
    else
    {

```

```

    minutes += int(deg) % 100;
    deg = int(deg) / 100;
}

// add minutes to degrees
deg=deg+minutes/60;
if (neg == true)
{
    deg*=-1.0;
}
neg = false;

if( deg < 0 ){
    neg = true;
    deg*=-1;
}
float numeroFloat=deg;
int parteEntera[10];
int cifra;
long numero=(long)numeroFloat;
int size=0;
while(1){
    size=size+1;
    cifra=numero%10;
    numero=numero/10;
    parteEntera[size-1]=cifra;
    if (numero==0){
        break;
    }
}
int indice=0;
if( neg ){
    indice++;
    input[0]='-';
}
for (int i=size-1; i >= 0; i--)
{
    input[indice]=parteEntera[i]+'0';
    indice++;
}
input[indice]='.';
indice++;
numeroFloat=(numeroFloat-(int)numeroFloat);
for (int i=1; i<=6 ; i++)
{
    numeroFloat=numeroFloat*10;
    cifra= (long)numeroFloat;
    numeroFloat=numeroFloat-cifra;
    input[indice]=char(cifra)+48;
    indice++;
}
input[indice]='\0';
}

void send_HTTP(String temps){
    String str = temps;
    // Length (with one extra character for the null terminator)
    int str_len = str.length() + 1;
    // Prepare the character array (the buffer)
    char char_array[str_len];
    // Copy it over
    str.toCharArray(char_array, str_len);
    // Initializes HTTP service

```

```

answer = sendATcommand("AT+HTTPINIT", "OK", 10000);
if (answer == 1)
{
    // Sets CID parameter
    answer = sendATcommand("AT+HTTTPARA=\"CID\\", 1", "OK", 5000);
    if (answer == 1)
    {
        // Sets url
        sprintf(aux_str, "AT+HTTTPARA=\"URL\\", \"http://%s/demo_sim908.php?\", url);
        Serial.print(aux_str);
        sprintf(frame,
"visor=false&id=5&glucosa=%s&latitude=%s&longitude=%s&altitude=%s&time=%s&satellites=%s&speedOTG=%s",
        char_array, latitude, longitude, altitude, date, satellites, speedOTG);
        Serial.print(frame);
        answer = sendATcommand("\\\"", "OK", 5000);
        if (answer == 1)
        {
            // Starts GET action
            answer = sendATcommand("AT+HTTPACTION=0", "+HTTPACTION:0,200", 30000);
            if (answer == 1)
            {
                Serial.println(F("Done!"));
            }
            else
            {
                Serial.println(F("Error getting url"));
            }
        }
        else
        {
            Serial.println(F("Error setting the url"));
        }
    }
    else
    {
        Serial.println(F("Error setting the CID"));
    }
}
else
{
    Serial.println(F("Error initializing"));
}

sendATcommand("AT+HTTPTERM", "OK", 5000);
}

```

```

int8_t sendATcommand(char* ATcommand, char* expected_answer1, unsigned int timeout){

```

```

    uint8_t x=0, answer=0;
    char response[100];
    unsigned long previous;

    memset(response, '0', 100); // Initialize the string

    delay(100);

    while( Serial.available() > 0) Serial.read(); // Clean the input buffer

```

```

Serial.println(ATcommand); // Send the AT command

    x = 0;
    previous = millis();

    // this loop waits for the answer
    do{
        if(Serial.available() != 0){
            response[x] = Serial.read();
            x++;
            // check if the desired answer is in the response of the module
            if (strstr(response, expected_answer1) != NULL)
            {
                answer = 1;
            }
        }
        // Waits for the answer with time out
    }
    while((answer == 0) && ((millis() - previous) < timeout));

    return answer;
}

String glucosa()
{
    uint8_t numberOfData = eHealth.getGlucometerLength();
    Serial.print(F("Number of measures : "));
    Serial.println(numberOfData, DEC);
    delay(100);
    String temps;

    for (int i = 0; i<numberOfData; i++) {
        Serial.println(F("====="));
        Serial.print(F("Measure number "));
        Serial.println(i + 1);
        Serial.print(F("Date -> "));
        Serial.print(eHealth.glucoseDataVector[i].day);
        Serial.print(F(" of "));
        Serial.print(eHealth.numberToMonth(eHealth.glucoseDataVector[i].month));
        Serial.print(F(" of "));
        Serial.print(2000 + eHealth.glucoseDataVector[i].year);
        Serial.print(F(" at "));
        if (eHealth.glucoseDataVector[i].hour < 10) {
            Serial.print(0);
        }

        Serial.print(eHealth.glucoseDataVector[i].hour);
        Serial.print(F(":"));

        if (eHealth.glucoseDataVector[i].minutes < 10) {
            Serial.print(0);
        }
        Serial.print(eHealth.glucoseDataVector[i].minutes);

        if (eHealth.glucoseDataVector[i].meridian == 0xBB)
            Serial.println(F(" pm"));
        else if (eHealth.glucoseDataVector[i].meridian == 0xAA)
            Serial.println(F(" am"));
    }
}

```

```

        Serial.print(F("Glucose value : "));

        Serial.print(eHealth.glucoseDataVector[i].glucose);

        temps = temps+"-" + eHealth.glucoseDataVector[i].glucose;

    }

    return temps;

}

```

Fuente: Elaboración propia con base en Libelium (s.f.)

Recepción de los datos. Teniendo en cuenta que la arquitectura a implementar es un Sistema cliente –servidor, a continuación se hace referencia el script que permite tomar los datos enviados desde el dispositivo móvil, el cual a través de una red de datos y el protocolo de aplicación http, envía los datos necesarios para su ubicación junto con los otros censados del glucómetro digital, este script desarrollado en *PHP* permite a través de la función *GET*, obtener los respectivos datos, entre estos tenemos los datos de geo posicionamiento, latitud, longitud, velocidad y curso, los cuales están representados por una variable en cada transacción de los datos en el sistema, los otros datos relevante son las medidas censadas por el glucómetro digital, cada medida acompañada por su fecha y hora obtenida, no el tiempo en que se realizó la transacción sino el tiempo y fecha en que se censó la variable, las cuales pueden ser de 1 a varias, es decir, el paciente puede realizar varias mediciones con el dispositivo digital durante el día, y en 1 sola transacción puede enviar todas las variables censadas junto con su tiempo en que fue tomada.

```
<?php
```

```

$lat = getParameter("latitude");
$lon = getParameter("longitude");

```

```

$hostname = "yair.hostingnubedigital.com";
$username = "yair_costac";
$dbname = "yair_costac";

//These variable values need to be changed by you before deploying
$password = "Y00GBAG32c!";
$usertable = "prueba";
$yourfield = "nombre";

$idpaciente = getParameter("id");
$longitud = $lon;
$latitud = $lat;
$glucosa = htmlspecialchars($_GET["glucosa"], ENT_QUOTES);

echo "Glucosa: ".$glucosa;

$url = explode("-", $glucosa);

foreach ($url as $valor) {

    mysql_connect($hostname, $username, $password) OR DIE ("Unable to
    connect to database! Please try again later.");
    mysql_select_db($dbname);

    $query = " insert into controlpaciente(fecha,idpaciente,longitud,latitud,glucosa,hora) values (NOW(),'$idpaciente', '$longitud',
    '$latitud', '$valor', CURRENT_TIMESTAMP())";
    $result = mysql_query($query);
    echo $query;
}

/*if ($result) {
    while($row = mysql_fetch_array($result)) {
        $name = $row["$yourfield"];
        echo "Name: $name<br>";
    }

} */

if (!file_put_contents($file, $person, FILE_APPEND | LOCK_EX))
    echo "\n\t Error saving Data\n";
else echo "\n\t Data Save\n";

```



```

    }
    else {

?>

<!DOCTYPE html>
<html>

<head>

    <!-- Load JQuery -->

    <script language="JavaScript" type="text/javascript" src="jquery-1.10.1.min.js"></script>

    <!-- Load Google Maps Api -->

    <!-- IMPORTANT: change the API v3 key -->

//    <script src="http://maps.googleapis.com/maps/api/js?key=your_key&sensor=false"></script>

    <script src="https://maps.googleapis.com/maps/api/js?v=3.exp"></script>

<!-- Initialize Map and markers -->

    <script type="text/javascript">
        var myCenter=new google.maps.LatLng(41.669578,-0.907495);
        var marker;
        var map;
        var mapProp;

        function initialize()
        {
            mapProp = {
                center:myCenter,
                zoom:15,
                mapTypeId:google.maps.MapTypeId.ROADMAP
            };
            setInterval('mark()',5000);
        }

        function mark()
        {
            map=new google.maps.Map(document.getElementById("googleMap"),mapProp);
            var file = "gps.txt";
            $.get(file, function(txt) {
                var lines = txt.split("\n");
                for (var i=0;i<lines.length;i++){
                    console.log(lines[i]);
                    var words=lines[i].split(",");
                    if ((words[0]!="")&&(words[1]!=""))
                    {
                        marker=new google.maps.Marker({
                            position:new google.maps.LatLng(words[0],words[1]),
                        });
                        marker.setMap(map);
                        map.setCenter(new google.maps.LatLng(words[0],words[1]));
                        document.getElementById('sat').innerHTML=words[3];
                        document.getElementById('speed').innerHTML=words[4];
                        document.getElementById('course').innerHTML=words[5];
                    }
                }
            })
            marker.setAnimation(google.maps.Animation.BOUNCE);
        }
    </script>

```

```

        });
    }

    google.maps.event.addDomListener(window, 'load', initialize);
</script>
</head>
<body>
    <?php
        echo '

        <!-- Draw information table and Google Maps div -->

        <div>
            <center><br />
                <b> SIM908 GPS position DEMO </b><br /><br />
                <div id="superior" style="width:800px;border:1px solid">
                    <table style="width:100%">
                        <tr>
                            <td>Time</td>
                            <td>Satellites</td>
                            <td>Speed OTG</td>
                            <td>Course</td>
                        </tr>
                        <tr>
                            <td id="time">'. date("Y M d - H:m") .'</td>
                            <td id="sat"></td>
                            <td id="speed"></td>
                            <td id="course"></td>
                        </tr>
                    </table>
                </div>
                <br /><br />
                <div id="googleMap" style="width:800px;height:700px;"></div>
            </center>
        </div>';
    ?>
</body>
</html>

<?php } ?>

```

Fuente: Elaboración propia con base en Google Developed, Google Maps (s.f.)

El script del servidor se conecta a una base de datos geográfica hecha en mysql, a través de los parámetros: nombre del servidor, nombre de usuario y contraseña, para posteriormente a través de una instrucción *SQL*, guardar los datos recibidos por el script y censados desde el origen de la comunicación, además facilita de alguna manera, el acceso a recursos de índole espacial, esto según Stones y Matthew (2005,p 124-136), el portal permite además:

- Integra todas las diferentes fuente de la información
- Acceso directos de los datos
- Mejora la toma de decisiones
- Facilita el manejo espacial y cartográfico de los datos.
- Colabora y comparte información

Arquitectura de servicios espaciales. Esta parte del sistema permite manejar todo lo referente a la ubicuidad del paciente a través del almacenamiento y consulta de los datos geoespaciales en una arquitectura multicapa, la cual es la plataforma de información, o la arquitectura de datos que da soporte al script que corre en el servidor.

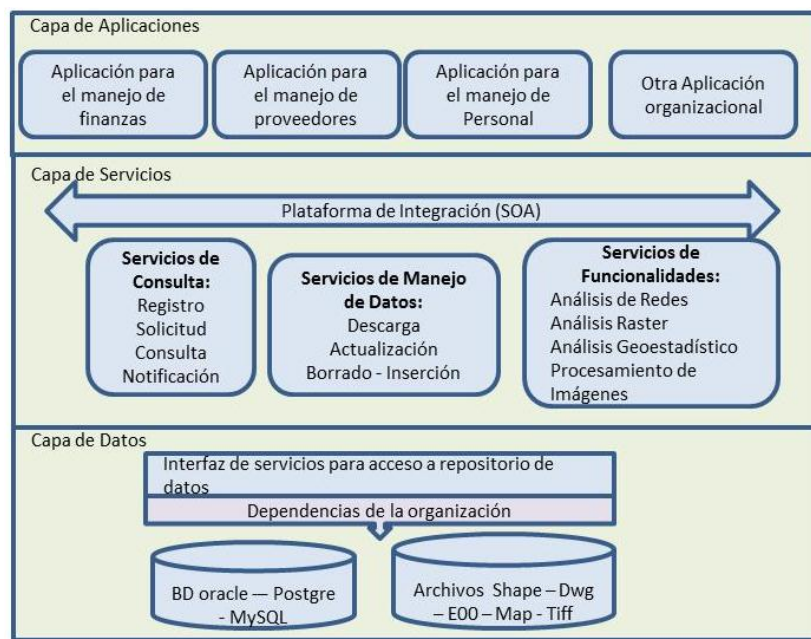


Figura 7. Arquitectura de la plataforma SIG(Pardo y Pardo,2011, p 44)

Según Pardo y Pardo (2011,) Una arquitectura que para su implementación, necesita distribuirse en varias capas.

Capa de aplicación: Se determinan una serie de aplicaciones necesarias e independientes entre sí, pero que juntas atienden a los requerimientos funcionales, estas son necesarias para el intercambio de datos, aquí la conexión se implementa teniendo como base central los servicios

web, los cuales dan una comunicación entre los usuarios a través de la interfaz de la plataforma, dándose una integración entre el envío y la recepción de los datos.

Capa de servicios: Contiene una gran variedad de servicios cartográficos y espaciales todos relacionados con los procesos de consulta y análisis de los datos en el sistema, los cuales poseen una interdependencia a nivel de la capa de aplicación, pero que relacionan todos a un mismo nivel en esta capa, Se incrusta toda una gran variedad de clases de consultas espaciales, así como una serie de funciones de análisis compleja de datos.

Capa de acceso a datos: Se incluye la consolidación de los datos, los cuales se presentan al usuario como un ente centralizado pero que en realidad pueden estar distribuidos en cualquier parte de la organización, esta sección se puede optimizar haciéndola totalmente dinámica y eficiente, a través del manejo de hilos múltiples para el acceso a datos.

Flujograma de Servicios

El dispositivo parte de una adaptación básica de un modelo de comunicación en telemedicina, el cual cuenta con una unidad emisora y otra receptora o software corriendo en un servidor. La unidad o el dispositivo emisor esta preconfigurado de tal manera, que el paciente diabético pueda ser identificado a través de la información registrada en el *SIG*, la cual constantemente monitorea sus niveles de glucosa, al mismo tiempo dado que los especialistas médicos están geográficamente ubicados en diferentes zonas o centros de atención especializada, es prioritario que la unidad envíe toda una información consolidada, referente a la ubicación geográfica del paciente, para precisar su ubicuidad y nivel de glucosa, junto con la fecha y hora de toma de los datos, con esto la unidad receptora o servidor, conoce de manera instantánea cada valor del sistema, digitalizando estos valores en el sistema, se procede a difundir la información

entre los medios interesados. Posteriormente después de que la información es procesada, el paciente es valorado por el médico enlazado a su historia clínica y si es necesario podría realizarse de manera instantánea una atención inmediata a través del centro más cercano o ambulancia más cercana.

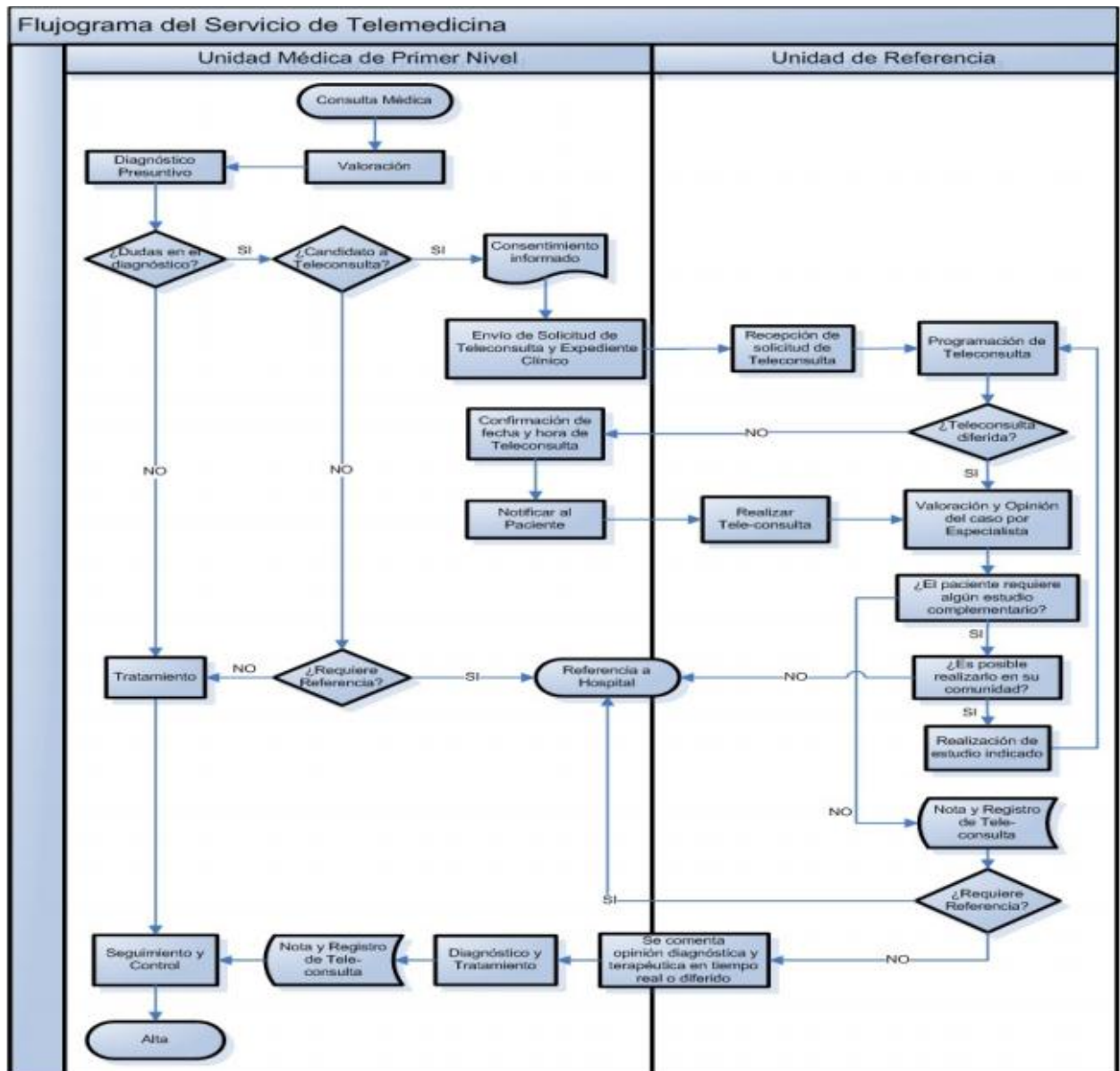


Figura 8. Flujo grama de telemedicina Móvil Fuente: Centro Nacional de Excelencia Tecnológica en Salud CENETEC (20 de enero de 2011, p. 27)

Para el ahorro de costos e infraestructura física el sistema de información debe estar montado en la web, la cual de igual forma hace parte de la infraestructura básica para la prestación de servicios médicos, ayudando directamente a la prestación eficiente de servicios médicos con la gestión y atención permanente de pacientes diabéticos.

Modelado del sistema. El modelado del sistema se realizará a través de la metodología unificada de modelado (UML), esta metodología es un estándar para modelar tecnología orientada objeto, nos presenta una descripción de alto nivel, donde nos describen, ya más detalladamente cada caso de uso definido por los usuarios, y la forma en que ellos interactúan con el sistema, llamándose a cualquier usuario del sistema “Actores”, los cuales se encuentran identificados en un modelo genérico, para este caso podríamos determinar un modelo básico acorde a Grover (2012).

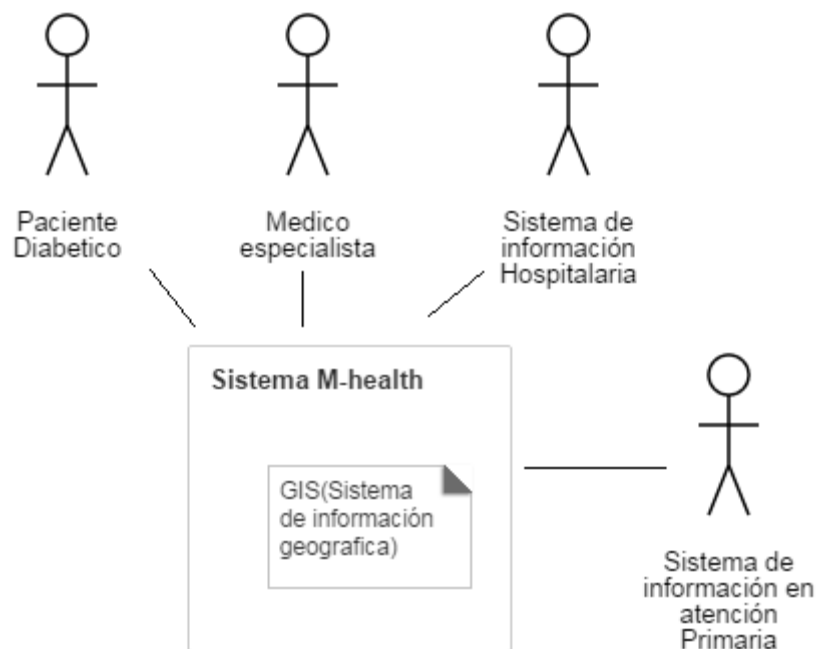


Figura 9. Sistema de información de telemedicina Móvil M-Health(Msc Yair Rivera,2014).

Este modelado general incluye tanto la infraestructura física como los actores que intervienen en ella, en este caso el paciente, los centros de control y médicos o especialista, el modelo central o de telemedicina móvil “*M-health*” a la vez la componen el sistema de monitoreo de niveles de glucosa y el sistema de información Geográfica.

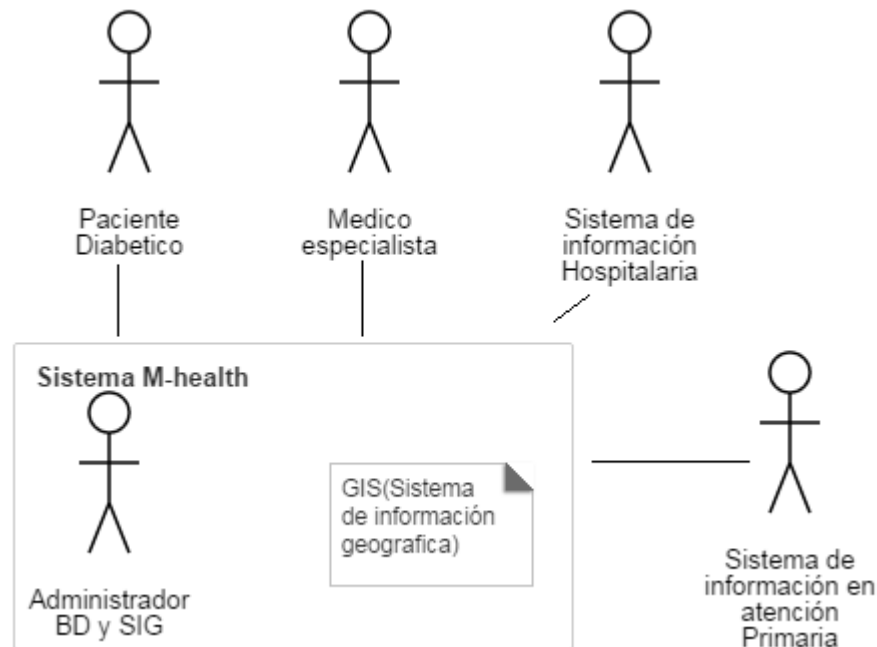


Figura10. Sistema modular del M-Health y el SIG(Msc Yair Rivera,2014).

El monitoreo y seguimiento de cualquier paciente ha de presentarse de forma integrada, debe ser percibido como un solo sistema y no por separado, es decir el geo posicionamiento o tracking de cada individuo debe ir relacionado con las lecturas típicas del dispositivo, de ahí que el caso de uso de la anterior figura, implementa directamente los servicios e integrando los 2 servicios, con el único propósito de mostrarlos como un solo sistema, esto siguiendo a Marquillas, Brú, Bertet e Ibáñez (2013).

Para Hababeh, Khalil, y Khreishah, A. (2014), podríamos identificar la funcionalidad del sistema a la hora de establecer una comunicación, esta podría ser de modo asincrónico a través del envío y almacenamiento de los datos, o uno modo sincrónico, esto último enviando el dispositivo mensajes de alarmas a través del sistema *GPRS*, todo este funcionamiento comprenden distintos niveles de priorización y como consecuencia de atención sanitaria, se trata de la comunicación full dúplex entre pacientes , médicos y usuarios del sistemas o tele operador, refiriéndonos a este último como el manager o administrador de la base de datos. Ver diagrama caso de uso en la siguiente figura.

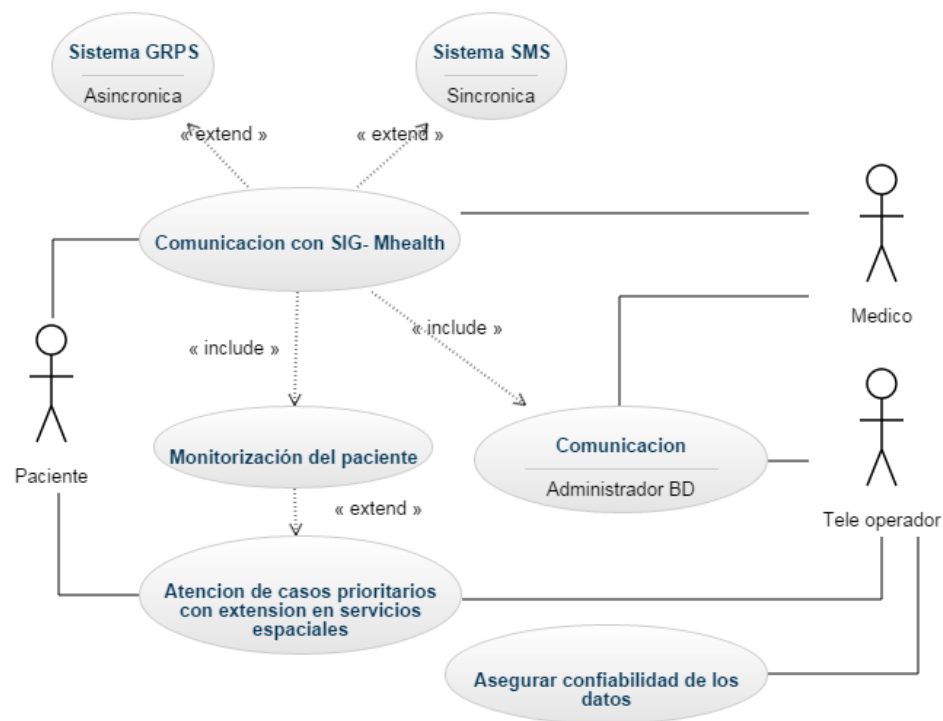


Figura 11. Diagrama caso de uso del sistema (Msc Yair Rivera,2014).

La monitorización del paciente se da a través de la toma de las muestras a través del glucómetro digital, en este punto el dispositivo selecciona que tipo de comunicación procesar, depende de las lecturas leídas y digitalizadas en el sistema, este puede procesar un almacenamiento normal de

datos a través de internet, o generar *SMS* dentro de la red celular también, con el objetivo de iniciar una atención sanitaria con prioridad.

Implementación y almacenamiento. Con base a Wong (2000), para que el dispositivo móvil tenga conexión con la base de datos a través de una red de datos, se implementó el protocolo de comunicación o de transferencia de Datos de hipertexto *HTTP*, el cual nos permite realizar transferencias de datos a través del internet a nivel de aplicación y de forma distribuida, para que la comunicación sea exitosa, es necesario tener la ubicación del servidor de datos en la internet a través de un identificador nombres *URL*, que para este caso sería: "*yair.hostingnubedigital.com/telemedicina/demo_sim908.php*". Este identificador de recursos nos permitirá implementar una arquitectura de comunicación Cliente-Servidor., donde el protocolo *HTTP* se asienta en el paradigma de solicitud/respuesta llevado través de un estado sin conexión, es decir después que el servidor ubicado en la web haya atendido cada petición realizada por el prototipo, la comunicación automáticamente se termina la conexión. Ver siguiente figura.

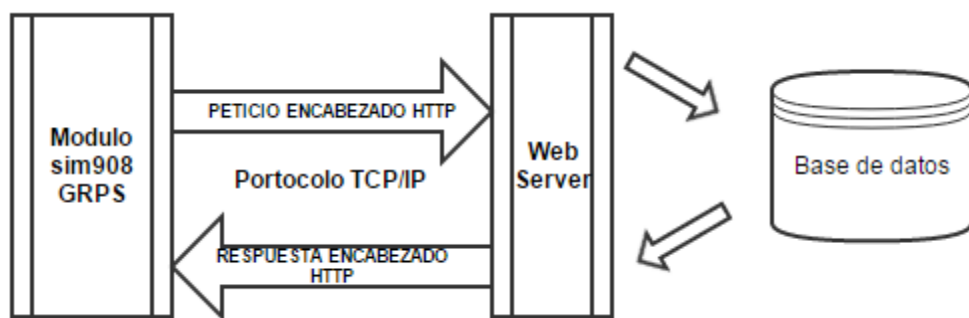


Figura 12. Envío de la información (Msc Yair Rivera,2014).

Para enviar peticiones a través del protocolo *HTTP* es necesario la codificación de encabezados, el cual es un bloque de datos que se hace referencia a él como metadatos, la comunicación estaría dada de la siguiente forma:

El prototipo realizado realiza una petición *HTTP*, con la información a enviar en este caso: Mediciones digitalizadas, y posición actual del paciente

El servidor web procesa la solicitud a través de un script alojado en el servidor web, toma los datos y lo almacena en la base de datos, luego el script le envía una respuesta *HTTP*.

Realmente el proceso de la comunicación es más extenso si consideramos todo el procesamiento de la transacción, pero para ser más centrados solo se estudió el procesamiento del protocolo *HTTP*, empezando con peticiones o tipo de documento solicitado, que consta de 3 líneas, el método, la dirección *URL* y la versión del protocolo, *HTTP* define 8 métodos que indican las acciones a ejecutar sobre recursos del servidor, es decir para arquitectura un script corriendo en el servidor, a continuación se hace referencia a los métodos más importantes:

GET. Como parámetro tiene una *URL* del recurso compartido, donde al mismo tiempo puede enviar variables para compartir con el otro extremo de la comunicación, estas concatenadas con ampersands (&).

Ejemplo: `http://www.paginaservidor.com/index.php?page=mia&var1=valor2&variable3=valor3..`

HEAD. Al realizar una petición cualquiera sobre el servidor, este comando nos sirve para obtener su cabecera de respuesta, su meta-información escrita en esta, importante para saber si ciertos recursos se encuentran ubicados en el servidor.

POST. Da la orden al servidor de esperar una información enviada desde el cliente, en este caso los datos son enviados desde el cuerpo de la información, y permite trabajar en la capa

persistente del software, es decir las operaciones básicas de una Base de datos, CRUD (Crear, Obtener, Actualizar y Borrar)

PUT. Permite escribir un archivo a través de una conexión socket creada en el servidor, es decir envía un recurso especificado desde el cliente hasta el servidor.

OPTIONS. De gran importancia para obtener información de las características de la sesión establecidas por la comunicación entre el cliente y el servidor, es decir un método importante para conocer y setear los parámetros de la comunicación.

Para este proyecto se utilizara el método GET, y tiene como parámetro la URL del recurso dentro del servidor, unida a las variables a enviar por el cliente, ver el siguiente código.

```
sprintf(aux_str, "AT+HTTPPARA=\\\"URL\\\",\\\"http://%s/demo_sim908.php?\", url);
Serial.print(aux_str);
sprintf(frame,
"visor=false&id=5&glucosa=%s&latitude=%s&longitude=%s&altitude=%s&time=%s&satellites=%s&speedOTG=%s",
char_array,latitude, longitude, altitude, date, satellites, speedOTG);
Serial.print(frame);
answer = sendATcommand("\\\"", "OK", 5000);
```

Fuente: Elaboración con base en Libelium (s.f.)

En este caso las variables enviadas desde el dispositivo móvil son:

Id: Identificador del dispositivo.

Glucosa: es una cadena de caracteres comprendida por las lecturas respectivas del dispositivo móvil, junto con su fecha y hora obtenida.

Latitud, longitud, satélites y speedOTG: Variables tomadas desde módulo *GSM* del dispositivo móvil, el cual nos permite la comunicación *GRPS* con la red de datos de la red celular y obtener de manera instantánea la ubicación exacta, esto gracias a la red de satélites geoestacionarios, estas variables aparte de darnos la ubicación, nos permite manejar la velocidad y el curso del dispositivo, esto siguiendo a Rivera (2015, p. 36).

Teniendo construida la *URL* como parámetro principal para el envío de información, se procede a enviar la información, ver el siguiente código.

```
answer = sendATcommand("AT+HTTPACTION=0", "+HTTPACTION:0,200", 30000);
```

El comando *AT+HTTPACTION*, es utilizado para una comunicación exclusiva con la red de datos a través del protocolo *HTTP*, y el método *GET* en el envío de la información, la cual es tomada desde el otro extremo de la comunicación por un servidor web, a través de un script desarrollado en PHP donde se separa cada variable del sistema, ver siguiente código.

```
if (!empty($_GET['latitude']) && !empty($_GET['longitude']) &&
    !empty($_GET['time']) && !empty($_GET['satellites']) &&
    !empty($_GET['speedOTG']) && !empty($_GET['course'])) {
```

Fuente: Elaboración propia.

A través de la función *GET* para la obtención de variables enviadas a través de la *URL*, podemos separar cada variable y darle así un trato exclusivo dentro del sistema, finalmente tomamos el código donde se trabaja la itinerancia del sistema, ver siguiente código:

```
mysql_connect($hostname, $username, $password) OR DIE ("Unable to
connect to database! Please try again later.");
mysql_select_db($dbname);

$query = " insert into controlpaciente(fecha,idpaciente,longitud,latitud,glucosa,hora) values
(NOW(),'$idpaciente', '$longitud', '$latitud', ' $valor', CURRENT_TIMESTAMP());";
$result = mysql_query($query);
echo $query;
```

Fuente: Elaboración propia.

A través de una instrucción *SQL* se genera el almacenamiento de las variables correspondientes, en este caso se da un ejemplo básico de almacenamiento, la solución de la arquitectura se podría extender según las necesidades del sistema, teniendo en cuenta las normalizaciones básicas de la estructura de un modelo relacionar a seguir.

Base de datos geográfica y visualización. Para una arquitectura básica espacial, en el proyecto se implementa una base de datos open source *POSTGRES*, junto con un componente que maneja

los campos tipo espacial, llamado *POSTGIS*, el cual a la hora de crear las tablas o la estructura principal de la base de datos, nos permite hacer énfasis a la naturaleza del tipo de dato espacial geométrico, es decir, punto, línea, área. Para el proyecto es importante definir estas diferencias de tipo de datos, pues es aquí donde radican los servicios basados en la geometría espacial, es decir cómo brindar servicios basados en el geo posicionamiento del prototipo con referencias a infraestructuras físicas, esto siguiendo a Urbano y Cagnacci(2014), ya sean Hospitales, Ambulancias y Centros de atención, a continuación vemos un ejemplo de tabla para el almacenamiento de los puntos de ubicación, hay que resaltar que cada punto en el espacio, consta básicamente de 2 parámetros, latitud y longitud:

```
CREATE TABLE tracking
(
  tracking_id integer NOT NULL,
  date_collection date NOT NULL,
  date_insertion date NOT NULL,
  tracking_point geometry,
  shift_id integer DEFAULT 0,
  titulo text,
  descripcion text,
  color text,
  eventos character(150),
  direccion character(50),
  telefono numeric DEFAULT 0,
  celular numeric,
  imagen character(60),
  promociones character(150),
  CONSTRAINT tracking_pk PRIMARY KEY (tracking_id),
  CONSTRAINT enforce_dims_tracking_point CHECK (st_ndims(tracking_point) = 2),
  CONSTRAINT enforce_geotype_tracking_point CHECK (geometrytype(tracking_point) = 'POINT'::text OR
tracking_point IS NULL),
  CONSTRAINT enforce_srid_tracking_point CHECK (st_srid(tracking_point) = 32661)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE tracking OWNER TO postgres;
```

Fuente: Elaboración propia.

Teniendo en cuenta el concepto de Llario y Llario (2013), a diferencia de una estructura típica de almacenamiento, en aquellas de tipo espacial hay que relacionar el tipo del campo, su naturaleza geométrica espacial, esto se da a través de la instrucción: *geometrytype*, al mismo tiempo es necesario aclarar que estos comandos, solo hacen una estricta referencia sobre el establecimiento

de la geometría de la tabla o el campo, no hacen ninguna transformación de coordenadas o sistema de proyección, función que si es determinada por la instrucción `st_srid`, encargada de establecer el sistema espacial de referencia que para este caso sería el sistema espacial 32661 o WGS 84 North Meter, este sistema de referencia espacial es un sistema de coordenadas geodésicas o coordenadas geográficas, que asume unas medidas definidas para la geografía de la tierra, en pocas palabras un plano cartesiano geográfico. Ver siguiente figura:

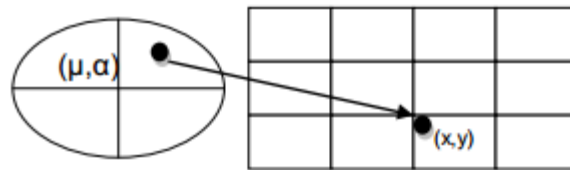


Figura13. Representación geográfica de coordenadas elipsoides.

El sistema espacial WGS 84 (SRID: 4326) de referencia nos permite trabajar: Puntos, áreas, líneas y colecciones de geometrías definidas como multipuntos o multilíneas, y aplicar funciones de procesamiento para datos geográficos espaciales o procesamiento sobre la geodesia de la tierra: Conversiones a otros sistemas de referencias, Gestión del territorio, obtener medidas reales y generar geometría a partir de consultas espaciales.

Evaluación del Sistema. Para la prueba del dispositivo final se realizaron 3 mediciones reales sobre el glucómetro, tomadas a través de las láminas sensoras conectadas al mismo, consecutivamente se procedió a conectar el glucómetro con la plataforma *e-Health Sensor Shield*, encargada principalmente de procesar y digitalizar los datos. Ver siguiente figura:

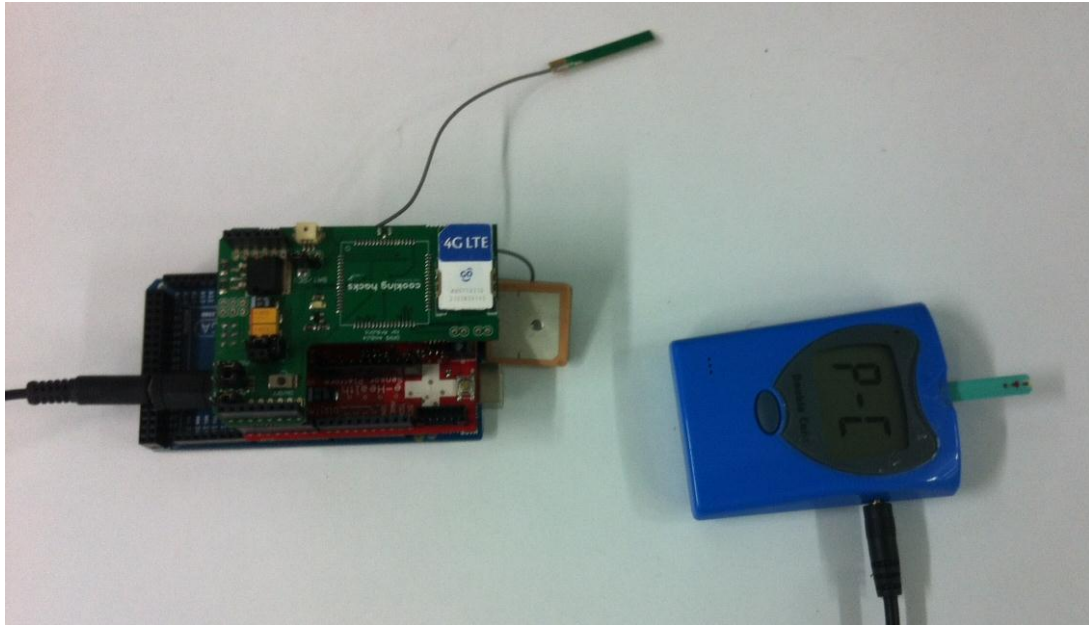


Figura 14. Dispositivo electrónico para recepción y envío de datos (Msc Yair Rivera,2014).

Los datos digitalizados, consisten en el número de mediciones tomadas, junto con la fecha y hora en que se realizó la medición, así como sus unidades respectivas de medición. Ver siguiente figura:

```

Number of measures : 3
=====
Measure number 1
Date -> 1 of January of 2009 at 01:11 am
Glucose value : 173 mg/dL
=====
Measure number 2
Date -> 1 of January of 2009 at 01:09 am
Glucose value : 93 mg/dL
=====
Measure number 3
Date -> 1 of January of 2009 at 01:02 am
Glucose value : 197 mg/dL

```

Figura 15. Digitalización de datos censados (Msc Yair Rivera,2014).

Digitalizados los datos dentro del sistema, el módulo *GSM* a través del *chip908* y la antena *GPS* procede a generar las coordenadas de geo referenciación, es decir latitud y longitud,

teniendo estos parámetros se procede a ingresar a la red de datos GPRS de la red celular, y a través del protocolo http se envía la información gracias al método de envío get, implementado en el lenguaje *AT ATCOMMAND*, ver siguiente figura.

```
COM3
Number of measures : 4
=====
Measure number 1
Date -> 3 of January of 2009 at 01:00 am
Glucose value : 137=====
Measure number 2
Date -> 1 of January of 2009 at 01:11 am
Glucose value : 173=====
Measure number 3
Date -> 1 of January of 2009 at 01:09 am
Glucose value : 93=====
Measure number 4
Date -> 1 of January of 2009 at 01:02 am
Glucose value : 197-137-173-93-197
Starting...
AT
AT+CPIN=****
AT+CGPSPMR=1
AT+CGPSPST=0
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CGPSSTATUS?
AT+CREG?
AT+SAPBR=3,1,"CONTTYPE","GPRS"
AT+SAPBR=3,1,"API","web.colombiamovil.com"
AT+SAPBR=3,1,"USER",""
AT+SAPBR=3,1,"PWD",""
AT+SAPBR=1,1
Starting GPS
AT+CGPSINF=0
FINISH GPS
AT+HTTPIPINIT
AT+HTTPPARA="CID",1
AT+HTTPPARA="URL","http://ysair.hostingmubedigital.com/telemedicina/demo_sim908.php?visor=false&cid=&glucosa=-137-173-93-197"
AT+HTTPACTION=0
```

Figura 16. Envío de datos censados (Msc Yair Rivera,2014).

Para este caso la información es enviada a un servidor web hostingweb.com, ver siguiente

código, Método de envío de la información:

http://hostingweb.com/telemedicina//demo_sim908.php?visor=false&id=5&glucosa=-173-93-197&latitude=10.991200&longitude=-74.807144&altitude=71

Finalmente la información es almacenada en tablas relacionadas y estructuradas, donde se relaciona la información con otras entidades del sistema, ver siguiente figura:

5	2014-12-09	15:38:05	173	10.9912	-74.807144
5	2014-12-09	15:38:05	93	10.9912	-74.807144
5	2014-12-09	15:38:05	197	10.9912	-74.807144

Figura 17. Datos almacenados en el sistema (Msc Yair Rivera,2014).

Toda esta integración de datos y servicios, da la posibilidad de implementar inteligencia en el tratamiento de la información y generar servicios especializados con base al manejo de espacios relativos, esto siguiendo a Anpeng,et al (2014, pp 320-416).

Criterios de selección de la tecnología. En general, el proceso de evaluación y selección de esta tecnología, coinciden en que los criterios de selección se fijaron en respuesta a las necesidades de los usuarios, dentro de los criterios más importantes para establecer criterios relevantes de diferenciación están:

	Arduino GSM	WIFI	Bluetooth
Estándares	GRPS	802.11	802.15
Velocidad	1.8 Mbps	54Mbps	700Kbps
Cobertura	Red celular	4-20m	30m
Movilidad	si	limitada	limitada
Espectro	licenciada	Uso común	Uso común
Precio	Moderado	Bajo	Bajo
Ubicuidad	Total	limitada	limitada
Seguridad	Encriptado	limitada	limitada

Tabla 1. Criterios de selección Arquitectura.

De acuerdo a siguiendo a Ikegami, Takeuchi, y Yoshida(1991, pp. 299–302), gracias a las características móviles de las redes de celulares *GSM*, esta permiten una alcance de una red *WAM* (Wide Area Network) o red de área amplia, lo que permite el envío de información a redes fuera del país, a diferencia de las tecnologías Wi-Fi y Bluetooth, los cuales necesitan los

servicios avanzados de una red de paquetes. La tecnología GSM trabaja en una gran amplia gama de espectro 850, 900, 1800 y 1900 MHz según Damosso (1998), lo que permite enviar la información a través de una plataforma menos sensible a errores de transmisión producida por cambios climáticos, obstáculos o generación aleatoria de ruido, y dinámicas en el tiempo, como atenuaciones, distorsiones y desvanecimientos de la señal, es decir una señal diseñada para trabajar con canales móviles de comunicación, la tecnología en que se desarrolló la plataforma nos permite tener además el mismo ancho de banda en diferentes áreas con una cobertura limitada por la red celular. Ver siguiente figura:

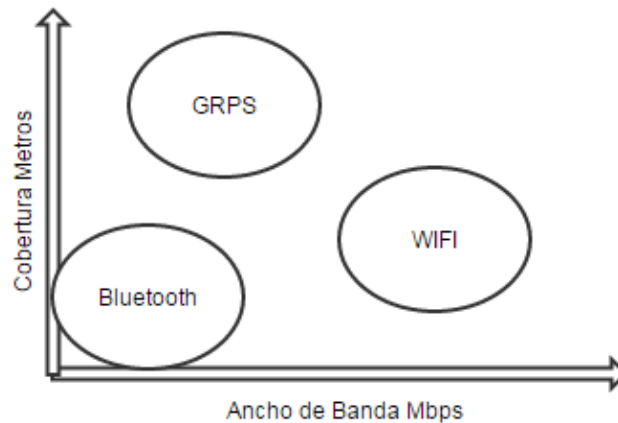


Figura 18. Criterio de selección Ancho de Banda-Cobertura (Msc Yair Rivera,2014).

Siguiendo a Zvonar, Jung y Kammerlande (2002), el ancho de banda considerado en la arquitectura telemática para la interconexión de los dispositivos móviles, es suficiente para controlar o medir en forma remota cualquier señal censada por el glucómetro digital, el nivel de señal captada por el dispositivo dota al paciente de una movilidad limitada por la red celular, al mismo tiempo la identificación del servicio estaría relacionado con un módulo suscriptor SIM (subscriber identity module) o tarjeta inteligente desmontable, el cual se inserta directamente en el módulo *GSM* arduino y que permite el encriptado transparente de la información con una llave

pública de *RSA* 4096 bit y una privada de 256 bit *AES*, la verdadera identificación del dispositivo móvil se realiza a través de esta tarjeta, ya que todos los servicios están asociados a ella.

En cuanto a la recepción física de la información estaría definida por la red típica de una arquitectura *GSM*, la arquitectura cuenta con una estación base o *BBS* (Station Subsystem), la cual permite conectar todos los dispositivos móviles asociados a ella con otra entidad más inteligente llamada *NSS* (Network and Switching Subsystem), esta última encargada de administrar las comunicaciones finales, y la cual a la vez consta de las siguientes entidades según Smith y Collins(2001):

- Mobile Services Switching Center (*MSC*): Es el centro del *NSS*, encargada de hacer las funciones de conmutación de la red, como de hacer las conexiones con otras redes externas.
- Gateway Mobile Services Switching Center (*GMSC*): es un gateway que permite realizar la comunicación entre las redes de telefonía fija y las móviles, este componente es necesario para enviar señales desde los teléfonos fijos a los prototipos móviles para telemedicina.
- Home Location Registrar (*HLR*): El *HLR* es una base de datos que contiene información sobre los pacientes conectados a una *MSC*, también tiene la información acerca de la localización del paciente en zonas específicas determinada por la triangulación de las antenas de la red celular, para este proyecto es fundamental este componente, ya que permite una ubicación del paciente *inDoor*, en la actualidad la ubicación del paciente está determinada por una antena GPS, lo que permite una ubicación *outDoor*.
- Visitor Location Registrar (*VLR*): Es una parte fundamental de la *HLR*, y es otra base datos donde relaciona a los pacientes con servicios específicos de acceso a la red de datos internet.
- Authentication Center (*AuC*): Maneja todo lo relacionado con la identificación y el acceso de la *SIM* a la red de conexión celular, es la parte que maneja la encriptación de esta conexión.

- GSM Interworking Unit (*GIWU*): sirve como interfaz de comunicación entre diferentes redes para comunicación de datos.

Finalmente la arquitectura telemática para telemedicina móvil, ver siguiente figura:

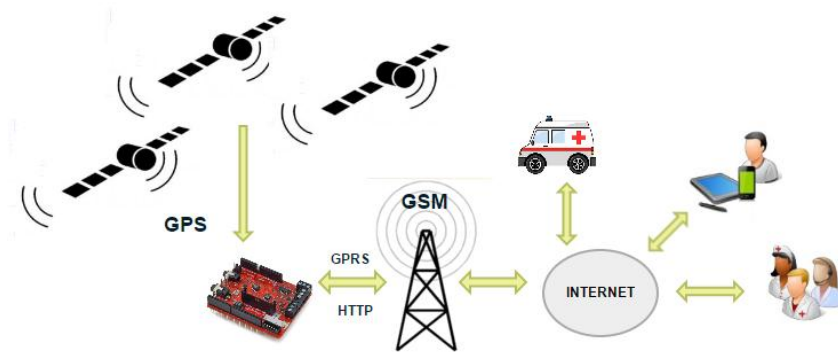


Figura 19. Arquitectura telemática M-health (Msc Yair Rivera, 2014).

Para Marrone (2004) , Khalil y Khreishah. (2014)., a través de la arquitectura se puede determinar el acceso a los dispositivos finales a través de la red de datos, se podrían acceder a los servidores de datos en la web, donde se almacena toda la información del sistema hospitalario, permitiendo de alguna manera acceder a esta información desde el prototipo móvil, de tal manera que cada prototipo estaría identificado, y con ello algunos servicios prioritarios estarían relacionados con servidores de aplicaciones y bases de datos, adicionalmente se podría resaltar la implementación de un eficiente sistema de seguridad que pueda establecer una sesión segura entre los dispositivos finales de la comunicación.

Resultados. Según Yang, et al(2014), la evidencia del logro se basa en un producto tecnológico, un prototipo electrónico inalámbrico con conexión vía *GRPS*, el cual a través de la red celular envía valores digitales como resultado de la lectura de un sensor glucómetro. Para ver

el dispositivo Ver figura 15, en la siguiente imagen se puede apreciar las diferentes tipos de lecturas digitales, tomadas junto con su hora y fecha obtenida por el sensor, Ver siguiente figura:

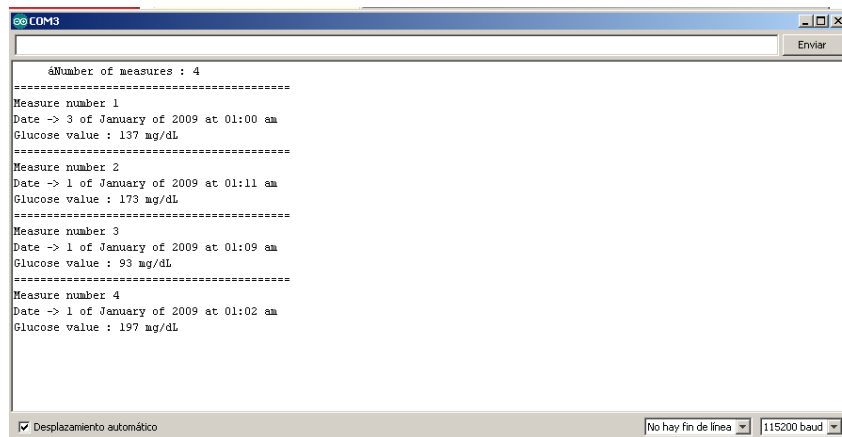


Figura 20. Valores digitalizados según lecturas del glucómetro digital(Msc Yair Rivera,2014).

Con base en Mukkundi, Bhattacharya, y Bhatt (2014), estos resultados se envían de forma instantánea hacia un sistema itinerante, lo que permite saber desde cualquier otro lugar del mundo, las lecturas tomadas, junto con la ubicación geográfica del paciente, dadas a través del módulo *GPS* del *SIM908*, un módulo que permite realizar conexiones a una red de datos a través de la red celular *GSM*, lo que permite de manera instantánea enviar los valores censados del nivel de glucosa en la sangre, junto con las coordenadas de geo referenciación, las cuales son receptadas por el servidor web y almacenadas por el GIS, consecutivamente estos valores pueden ser consultados con ayuda del API de google MAP. Ver siguiente figura:

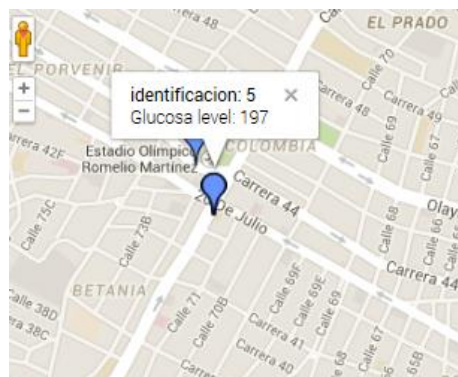


Figura 21. Consulta a través del API de google MAP(Msc Yair Rivera,2014).

La plataforma está diseñada para una arquitectura *IPv4*, para una fácil extensión a *IPv6* según el estándar 802.15.4, independientemente a esto, a nivel de la capa transporte del modelo *OSI*, toda la transmisión de la información se da a través del protocolo *UDP* (User Datagram Protocol), el cual es un protocolo de intercambio de datagramas que no requiere acuse de recibo para la transmisión de la información, lo que permite enviar pequeñas cantidades de datos de forma rápida y con menos consumo de energía, se resalta que el dispositivo es portable cuya energía esta suministrada por una batería de 5V, la tecnología de implementación para el envío de información tiene un bajo consumo, lo que lo hace operable por un largo tiempo sin reposición de batería.

Teniendo los datos en el sistema, a través de la herramienta *Postgis* de la base de datos *Postgres*, se habilita para ser usada como contenedor de información geo referenciada, permitiendo realizar operaciones de análisis gráfico, transformación de coordenadas, validación de datos con funciones para soportar almacenamiento de información raster, gestión de redes y cálculo de rutas, lo que lo hace todo un sistema de información geográfica sumado a los servidores de mapas de *google Map*, como fuente de cartografía digital, para desplegar visualmente todas las entidades del sistema.

Conclusión.

El diseño y programación de esta arquitectura telemática ha permitido conocer el potencial de los sistemas modulares en telemedicina *open source* basados en Arduino, relacionado al mismo tiempo con los sistemas de informaciones geográficas y los sistemas basados en la localización, con un procesamiento de la señal y actualización de datos en tiempo real, lo que permitió crear servicios en la web, basados en la geo referenciación del paciente e involucrando las infraestructuras físicas hospitalarias, es decir los servicios médicos geo referenciados, en este caso se asocia la localización del paciente con relación a la ubicación de cada infraestructura hospitalaria o servicio hospitalario, una arquitectura flexible con una cobertura limitada por la red celular permitiendo:

- Obtener un roaming internacional
- Aprovechamiento eficiente del espectro, a comparación con el envío de la información
- Velocidad en el envío y la actualización de los datos
- Portabilidad y movilidad apoyada en el protocolo IPv4 con extensión a IPv6
- Procesamiento y censado del nivel de glucosa de forma eficiente

Esta conectividad de extremo a extremo utilizando un direccionamiento global, permitió incorporar mecanismos de movilidad, con un bajo retardo en el procesamiento de las señales digitalizadas y tasas de transmisión causadas por la movilidad del cliente, sin embargo se facilita el envío y procesamiento de las señales, ya que los datos planos, y el protocolo de comunicación que se implementa, el protocolo *UDP* no es un protocolo robusto, permitiendo una comunicación así casi en tiempo real, Abonándole a esto último el sentido del proyecto, ya que los servicios de geo localización deben contar con una ubicación exacta del cliente y de las infraestructuras

hospitalarias, ayudando en gran medida a los pacientes crónicos, y a los enfermos en general, ya que aporta ventajas en cuanto a la monitorización de su enfermedad, los tratamientos a distancia o la tele rehabilitación.

Referencias

- Alotaibi, M. (2014). *An intelligent mobile diabetes management and educational system for Saudi Arabia: System architecture*; Medical Information and Network Technologies Research Centre, Kingston University, London, UK ; Istepanian, R.S.H. ; Sungoor, A. ; Philip, DOI N. 10.1109/BHI.6864296,2014, pp. 29–32.
- Anpeng, H., Chao, C., Kaigui, B., Xiaohui, D., Min, C., Hongqiao, G., Chao, M., Qian, Z., Yingrui, Z., Bingli, J. y Linzhen, X. (2014). *WE-CARE: An Intelligent Mobile Telecardiology System to Enable mHealth Applications. Biomedical and Health Informatics*, pp. 693-702.
- Centro Nacional de Excelencia Tecnológica en Salud CENETEC. (20 de enero de 2011).
- Clint Smith, Daniel Collins(2001), *3G Wireless Networks*,Edition McGraw-Hill telecommunications, ISBN
- Clinton Wong (2000),*HTTP Hypertext Transfer Protocol*, Edition O'Reilly & Associates, ISBN 1-56592-862-8, USA
- Daminni Grover (2012), *Object Oriented Analysis and Design with UML*,I.K. International Publishing House Pvt. Limited, 2012,ISBN
- Daojing, H., Chan, S., Yan, Z. y Haomiao, Y. (2014). *Lightweight and confidential data discovery and dissemination for wireless body area networks. Biomedical and Health Informatics*, pp. 440- 448.
- E. Damosso, editor, *Digital Mobile Radio: Towards Future Generation Systems*, (1998) European Commission, Final Report of the COST 231 Project. Chapter 4.

F. Ikegami, T. Takeuchi, and S. Yoshida, “*Theoretical prediction of mean field strength of urban mobile radio*, (vol. 39, pp. 299–302, March. 1991), IEEE Trans. Antennas Propagat.

Ferdinando Urbano, Francesca Cagnacci (2014)., *Spatial Database for GPS Wildlife Tracking Data*, Springer Cham Heidelberg New York Edition, ISBN 978-3-319-03742-4, DOI 10.1007/978-3-319-03743-1

Google Developed, Google Maps (S.F.), [Mensaje en un blog]. Recuperado de :<https://developers.google.com/maps/documentation/android/marker>

Hababeh, I., Khalil, I. y Khreishah, A. (2014). *Designing high performance web-based services to promote telemedicine database management system services computing*. IEEE Transactions on Digital Object Identifier: 10.1109/TSC.(2014)..2300499.

Janneth Pardo Pinzón, Rubby Stella Pardo Pinzón (2011), *Arquitectura basada en servicios para la integración de bases de datos espaciales*, Revista Azimut, 3, 39-46 .

José Martínez Llario, Juan Carlos Martínez Llario (2013). *PostGIS 2 análisis Espacial Avanzado*,

Josep Bras Marquillas, Josep Emili De la Flor Brú, María José Torregrosa Bertet, Itziar Martín Ibáñez (2013), *Pediatría en atención primaria + StudentConsult*, Tercera Edición, Elsevier Masson Edition, Barcelona-España, ISBN 978-84-458-2043-8,

Khalil, I. y Khreishah, A. (2014). *Designing high performance web-based computing services to promote telemedicine database management system*. IEEE transactions on volume.

Libelium (S.F.), Geolocation Tracker (GPRS + GPS) with SIM908

López, D. y Blobel, B. (2007). *Connecting public health and clinical information systems by using a standardized methodology*, (Pt. 1):132.

- Mahmoud, Y. (2010). *Information security strategy in telemedicine and e-health systems: A case study of England's shared electronic health record system*.
- Marrone, L. (2004). *Telemedicina sobre móvil ip (tesis de maestría)*. Universidad Nacional de la Plata, Argentina.
- Meulen, R. y Pettey, C. (2012). Gartner says world-wide sales of mobile phones declined 3 percent in third quarter of 2012; smartphone sales increased 47 percent. Recuperado de <http://www.gartner.com/newsroom/id/2237315>.
- Miller, M. (2011). *Using Google Maps and Google Earth*. USA: Pearson Education, Inc.
- Mukkundi, B., Bhattacharya, B. y Bhatt, O. (2014). *Design and development of a networked health monitoring and control system*, pp. 38–43, Print ISBN:978-1-4799-4381-4, Santa Clara, CA.
- Mykkänen, J., Korpela, M., Ripatti, S., Rannanheimo, J. y Sorri, J. (2007). *Local, regional and national interoperability in hospital-level systems architecture*. *Methods Inf Med*, 46(4), pp. 470-5.
- Over Arduino and Raspberry Pi [Mensaje en un blog]. Recuperado de: <http://www.cooking-hacks.com/documentation/tutorials/geolocation-tracker-gprs-gps-geoposition-sim908-arduino-raspberry-pi#http>
- Pérez, V., Martínez, J., Bustamante, J., Betancur, M. y Torres, O. (2011). *Sistema maestro-esclavo basado en dispositivos móviles para aplicaciones biomédicas*. En IV Congreso Colombiano de Bioingeniería e Ingeniería Biomédica.
- Rivera, Y. (2015). *Desarrollo de un prototipo Arduino-móvil y geoposicional en el área de la telemedicina para el monitoreo remoto de personas diabéticas a través de la red celular*. Barranquilla: Corporación Universitaria Americana.

- Richard Stones, Neil Matthew (2005), *Beginning Databases with PostgreSQL, From Novice to Professional*, Second Edition, ISBN 13-978-59059-478-0, USA
- Ruiz-Ibáñez, C. (2011). *Telemedicina: Historia, aplicaciones y nuevas herramientas en el aprendizaje*. Revista Ingeniería Biomédica, 6(11), pp. 10-21.
- Serie Tecnologías en Salud volumen 3. Telemedicina. Recuperado de <http://www.cenetec.salud.gob.mx/descargas/TecnologiasSaludV3.pdf>
- Simon Monk(2014), *Programming Arduino Next Steps: Going Further with Sketches*, McGraw-hill Education(Publisher), ISBN 978-0-07-183026-3, USA
- Yang, G., Xie, L., Mantysalo, M., Zhou, X., Pang, Z., Xu, L., Kao-Walter, S., Chen, Q. y Zheng, L. (2014). *A health-IoT platform based on the integration of Intelligent packaging, unobtrusive bio-sensor and intelligent medicine box*. Industrial Informatic.
- Z. Zvonar, Peter Jung, Karl Kammerlande (2002), *GSM: Evolution, Towards 3rd Generation Systems*, Kluwer Academic Publisher, ISBN 0-792-38351-6, New York, USA

