UNIVERSIDAD TECNOLOGICA DE TIJUANA

[](http://www.uttijuana.edu.mx/)

Tecnologías de la información y comunicación

OPTATIVA

Avance del proyecto: Rutas Inteligentes

Presentan:

Bermúdez Gradilla Ricardo

Montañez Lara Christian Ulises

Padilla Virgen José Luis

Orozco Sánchez José Carlos

Ramírez Higareda Jesús

Tijuana, Baja California, 06 de diciembre de 2016

ÍNDICE

[DESCRIPCIÓN DEL PROYECTO 5](#_Toc468820092)

[OBJETIVO GENERAL 6](#_Toc468820093)

[OBJETIVOS ESPECIFICOS 6](#_Toc468820094)

[ALCANCE 7](#_Toc468820095)

[JUSTIFICACIÓN 8](#_Toc468820096)

[ÁREA DE APLICACIÓN 9](#_Toc468820097)

[REQUERIMIENTOS FUNCIONALES 12](#_Toc468820098)

[RIESGOS DEL PROYECTO 13](#_Toc468820099)

[EXTERNOS 13](#_Toc468820100)

[TÉCNICOS 14](#_Toc468820101)

[EQUIPO 14](#_Toc468820102)

[CRONOGRAMA DEL PROYECTO 15](#_Toc468820103)

[Etapa de desarrollo 17](#_Toc468820104)

[DIAGRAMA DEL PROYECTO 18](#_Toc468820105)

[RECURSOS Y MATERIALES 19](#_Toc468820106)

[Implementación del sensor 19](#_Toc468820107)

[Programación del sensor con arduino 19](#_Toc468820108)

[Calibración del sensor 23](#_Toc468820109)

[Algoritmo sensor-Arduino 24](#_Toc468820110)

[Software 25](#_Toc468820111)

[Algoritmo de adquisición de datos Java-Arduino 25](#_Toc468820112)

[Diagrama de flujo 30](#_Toc468820113)

[Base de datos modelo E-R 31](#_Toc468820114)

[Almacenamiento de datos Java-Arduino 31](#_Toc468820115)

[Dashboard de grafica 36](#_Toc468820116)

[Resultados Obtenidos 37](#_Toc468820117)

[Pruebas 38](#_Toc468820118)

[Ilustración 1 Eficiencia del transporte 8](#_Toc468819554)

[Ilustración 2 Cronograma del Proyecto 15](#_Toc468819555)

[Ilustración 3 Cronograma del Proyecto 16](#_Toc468819556)

[Ilustración 4 Esquema del Proyecto 17](#_Toc468819557)

[Ilustración 5 Diagrama del Proyecto 18](#_Toc468819558)

[Ilustración 6 Modelo Entidad-Relación 31](#_Toc468819559)

[Ilustración 7 Dashboard 36](#_Toc468819560)

[Ilustración 8 Prueba: Obtención de datos vía modulo GPS 38](#_Toc468819561)

[Ilustración 9 Prueba: Envió de datos desde arduino hacia Java 38](#_Toc468819562)

**Etapa de planeación del proyecto**

Nombre del proyecto:

Rutas Inteligentes

# **DESCRIPCIÓN DEL PROYECTO**

El proyecto consiste en montar un dispositivo en unidades seleccionadas de transporte público (Principalmente en camiones los cuales fueron por los que optamos para empezar) para poder trazar la ruta que recorren, así como un tiempo estimado entre salida y llegada de un punto hacia otro. La implementación de este dispositivo determinara la ubicación en la que se encuentran y mandara la información recopilada hacia una base de datos donde se almacenaran.

Con el paso del tiempo este proyecto podría ser factible para remplazar a las personas que realizan el trabajo de tomar notas de los tiempos solamente que hacen de un punto hacia otro punto, acción que no es muy formal debido a que utilizan hojas y plumas y esto puede llevar problemas al momento de que se borren los datos y/o se pierda la hoja donde se realizaron los apuntes.

Por otra parte, sería una innovación para todos ya que esto hará que las ubicaciones sean más certezas y poder dar a los usuarios que estén esperando por el transporte que se le dé una hora estimada de la llegada entre camiones.

**OBJETIVOS**

**OBJETIVO GENERAL**

En estos tiempos no se sabe con precisión la hora en la que un transporte va a pasar y en muchas ocasiones llegamos tarde a nuestros compromisos aun cuando tomamos nuestro tiempo para llegar ya que no siempre se puede confiar mucho en el transporte, por lo tanto, la implementación de este proyecto será el poder conocer los tiempos que se tardan en realizar sus rutas y así mejorar este dato tanto como para usuarios del transporte, así como los empleados y jefes.

**OBJETIVOS ESPECIFICOS**

* Realizar el montaje del arduino, con todos los módulos correspondientes que se necesitan.
* Realizar comunicación con el servidor donde se almacenarán los datos obtenidos de las pruebas que se lleven a cabo.
* Una vez que esté terminado 100% el proyecto se instalara el sistema dentro del Transporte en el que vaya a ser utilizado dejándolo funcional para su uso diario.

# **ALCANCE**

Este dispositivo realizara la función de determinar la ubicación del transporte donde esté instalado al sistema eléctrico del mismo, con el fin de mandar datos al servidor, donde cada que se llegue de un punto A hacia un punto B lo cual permitirá a las personas que realicen la administración de esta información puedan realizar estimaciones previas, con el paso del tiempo ir generando gráficas mostrando los tiempos que se realizan durante todo el día entre otros datos que se puedan adquirir gracias a esta implementación como saber cuántos viajes llevan a cabo al día, a la semana, al mes.

Siendo esto una buena opción debido a que esto podría llegar a tener gran impacto sobre los conductores de las unidades, teniendo tiempos asignados cada una, no habría necesidad de ir a velocidades altas, interfiriendo dentro de los horarios de viajes de otras unidades, disminuyendo los accidentes por lo mismo de que hoy en día en la ciudad donde se vive los conductores de estas unidades no llevan un horario fijo y puede que haya manipulación de los “checadores” para apuntar horas inciertas y/o corrupción para dar datos falsos.

Entonces es ahí donde entra nuestro proyecto y su factibilidad debido a que serán datos precisos y no podrán ser alterados por personas con bajos conocimientos dentro del área tecnológica.

# **JUSTIFICACIÓN**

Según los reportes realizados dentro de la comunidad que hace uso de los transportes públicos en este caso camiones que son los que llevan más cantidad de usuarios, en veces es muy difícil el salir de su casa y que en cuanto vayan saliendo de la misma el transporte pase llevando esto a que la persona tenga una pérdida de tiempo considerable.

Esto no viene siendo problema del usuario del transporte, ni del conductor de la unidad si no la mala administración de tiempos de salida y de llegada, por si es cierto que se tiene un estimado por parte del usuario del transporte, pero aun así no es suficiente para que sea preciso.

Es ahí donde este proyecto entra en juego y marca la diferencia entre esos pequeños tiempos perdidos, dando así horarios más precisos para que el usuario esté preparado y listo para la llegada del transporte, así como también por parte del conductor de la unidad este a tiempo, el cual ya se haya estipulado y sea este un servicio más eficiente por parte de la empresa que adquiera este dispositivo.

Teniendo un aumento de eficiencia del transporte de un 20% dentro de los tres primeros meses, mostrando en la gráfica un ejemplo que de tardar 60 minutos en hacer su recorrido disminuiría a 48 minutos por la buena administración adquirida por este dispositivo.

Ilustración Eficiencia del transporte

# **ÁREA DE APLICACIÓN**

Este proyecto tiene claras ventajas dentro del área de transportes de la ciudad de Tijuana, Baja California. En donde no se lleva a cabo datos certeros, donde está limitado a los trabajadores internos de cada empresa para realizar la actividad de almacenar los datos en un cuaderno o carpeta.

Entonces es ideal la implementación de este proyecto dentro de esa área porque brindara la comodidad del cliente de obtener los datos con más precisión.

**FACTIBILIDAD ECONOMÍCA**

Este proyecto es factible porque es una forma sencilla la cual se realizará el trabajo, desde su instalación hasta el desarrollo del código. Todo esto gracias a que se utiliza software Open Source lo cual no requiere licencias para su uso.

El hardware utilizado es bajo costos y compatible con el software que se usara durante el proyecto, por lo tanto, económicamente puede ser implementado.

**FACTIBILIDAD TECNOLOGÍCA**

Este proyecto es factible porque tiene como base utilizar la tecnología Open source implementada en la tarjeta Arduino que es de un costo accesible para las empresas y usuario normal que se quiera dedicar a utilizar esta misma.

Por otra parte el uso de software igualmente Open Source el cual cada uno se puede adquirir fácilmente en sus páginas oficiales correspondientes.

**REQUERIMIENTOS TÉCNICOS**

Los conocimientos técnicos que se requiere en el personal que va implementar este proyecto sea un éxito y realizado a tiempo, son conocimientos en desarrollo web, conocimiento en el desarrollo de código en la plataforma de Arduino, conocimientos en electrónica y telecomunicaciones.

# **REQUERIMIENTOS FUNCIONALES**

|  |  |
| --- | --- |
| Requerimiento | Descripción |
| RF1 | El proyecto contara con un módulo de GPS (Global Positioning System) |
| RF2 | El proyecto contara con una aplicación web |
| RF3 | La aplicación web permitirá mostrar las rutas en tiempo real |
| RF4 | La aplicación deberá ser accesible desde cualquier dispositivo movil |
| RF5 | La aplicación deberá estar en funcionamiento el 99% del tiempo. |

# **RIESGOS DEL PROYECTO**

|  |  |  |
| --- | --- | --- |
|  | **EXTERNOS** |  |
| TIPO DE RIESGO | SOLUCIÓN | CAUSA |
| Entrega de material retrasada o perdida. | Tener uno o dos proveedores extras. | Material en mal estado o no entregada a tiempo. |
| Entrega de material no solicitados. | Tener contacto con el proveedor y realizar el cambio a la brevedad. | Notas erróneas por parte de la empresa y/o del cliente. |
| Entrega del proyecto retrasada. | Realizar las actividades en menor tiempo, pero haciéndolas con calidad. | Falta de planificación por parte de los integrantes del proyecto. |
| Reuniones no productivas  (No lograr el objetivo de la reunión, no realizar avance del proyecto). | Realizar una minuta enviándose por correo previamente para que los integrantes tengan conocimiento de lo que llevara a cabo en la reunión. | Falta de comunicación por parte del líder del proyecto.  Inasistencias sin previo aviso y/ retrasos a la reunión por parte de algún integrante. |
| Duplicación de proyectos | Realizar una funcionalidad extra que destaque el proyecto para no verse opacado o repetitivo y haga esto una falta de interés por parte de los clientes. | Usar ideas muy genéricas que están dentro del mercado, hablar con personas poco confiables sobre el proyecto. |
| Actividades repetitivas que retrasen el proyecto. | Apoyarnos sobre la planificación del proyecto al pie de la letra para no interferir en las actividades de los demás o que se repitan. | Falta de comunicación dentro del equipo, desacuerdos, inconformidades. |

|  |  |  |
| --- | --- | --- |
|  | **TÉCNICOS** |  |
| TIPO DE RIESGO | SOLUCIÓN | CAUSA |
| Falta de algún componente | Adquirir componente a la brevedad posible. | Cambios imprevistos sobre el uso de componentes. |
| Fallas de hardware | Tener un remplazo de preferencia. | Defecto de fábrica. |
| Conexión fallida en los equipos de cómputo. | Utilizar siempre el equipo de cómputo en cual fueron realizadas las pruebas. | Falta de permisos en equipos de cómputo, para realizar la conectividad. |
| Incompatibilidad con software. | Utilizar versiones del software donde ya se hayan realizado pruebas y concluidas satisfactoriamente. | Incompatibilidad por versiones, superiores o inferiores. Software no compatible. |

|  |  |  |
| --- | --- | --- |
|  | **EQUIPO** |  |
| TIPO DE RIESGO | SOLUCIÓN | CAUSA |
| Bajo interés o interés nulo por parte de algún integrante. | Distribuir actividades tomando en cuenta aptitudes y actitudes de cada integrante. | Inasistencia cuando se distribuían actividades, mala actitud, inconformidad con el proyecto. |
| Falta de planificación. | Desarrollar mediante alguna aplicación o ya sea en el cuaderno un calendario con actividades y fechas de entrega. | Falta de comunicación por parte del líder del proyecto y sus integrantes. |
| Falta de dirección del proyecto. | Darnos apoyo, orientarnos en lo que sea necesario, comentarios positivos. | No cuenta con las aptitudes necesarias para ser un líder. |
| Conocimiento insuficiente sobre el proyecto y lo que se pretende lograr. | Hacer una investigación previa por cada integrante del equipo sobre lo que se va a realizar, utilizar e implementar. | Falta de atención en clases cuando se explican las actividades, poco interés sobre las materias. |

# **CRONOGRAMA DEL PROYECTO**

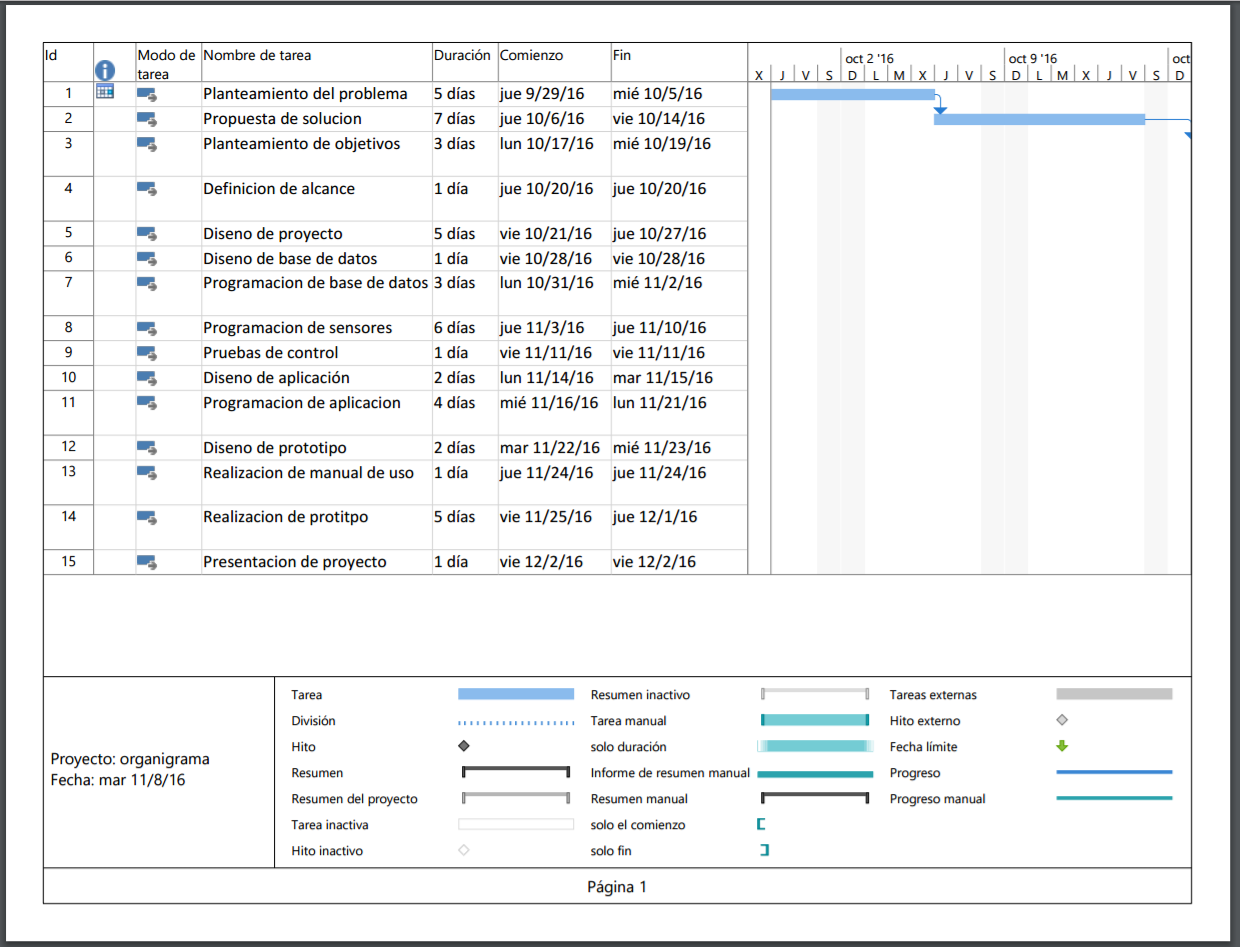
****

Ilustración Cronograma del Proyecto

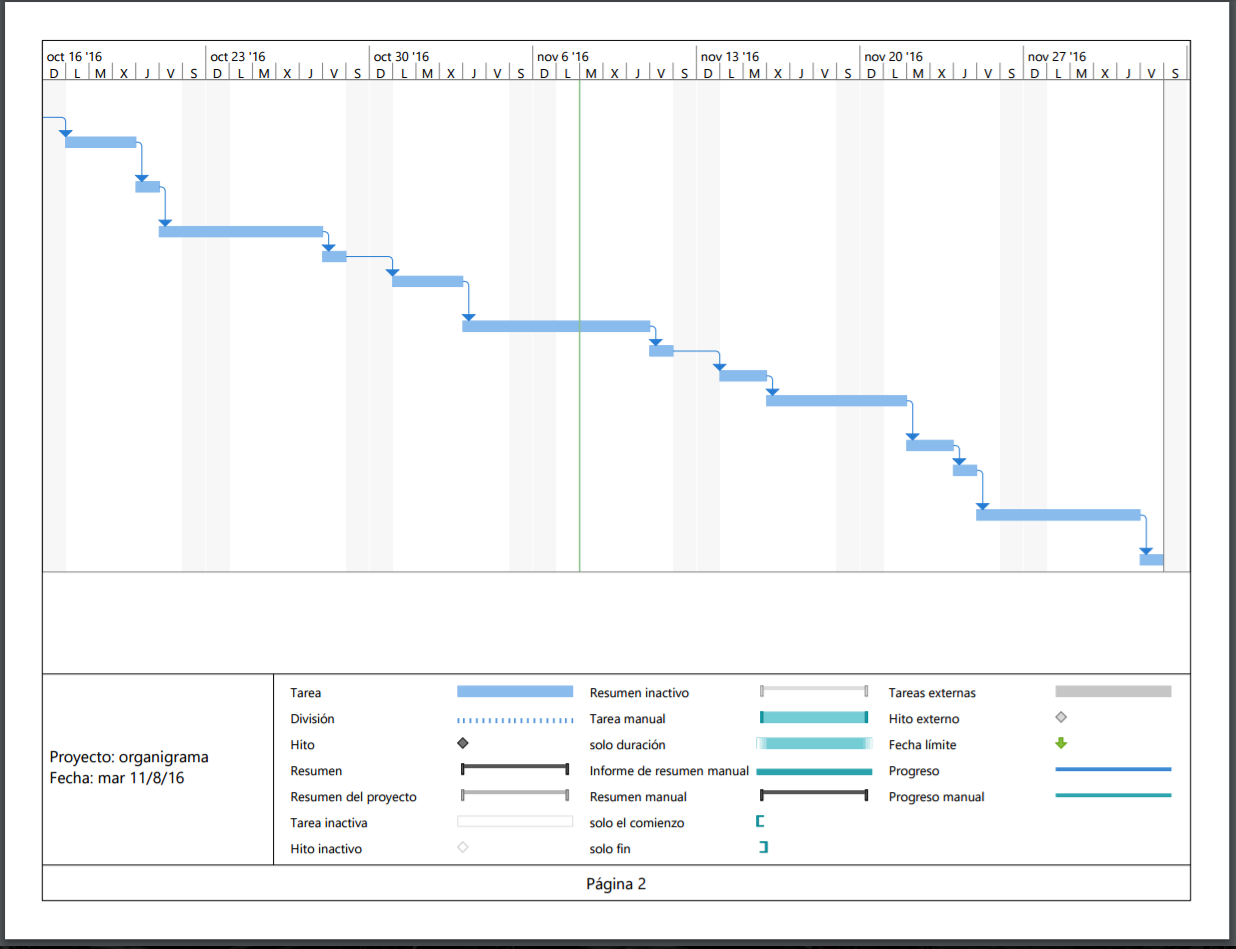
****

Ilustración Cronograma del Proyecto

# **Etapa de desarrollo**

**Hardware**

**Introducción**

Para la realización del proyecto, se implementa como hardware base la placa de programación Arduino, que, a su vez acorde a las necesidades de la solución, incluye el módulo de Ethernet Shield, para la implementación del módulo de comunicación GPS/GPRS para la transmisión de coordenadas en tiempo real de cada unidad portadora del mismo.

**Esquema del proyecto**

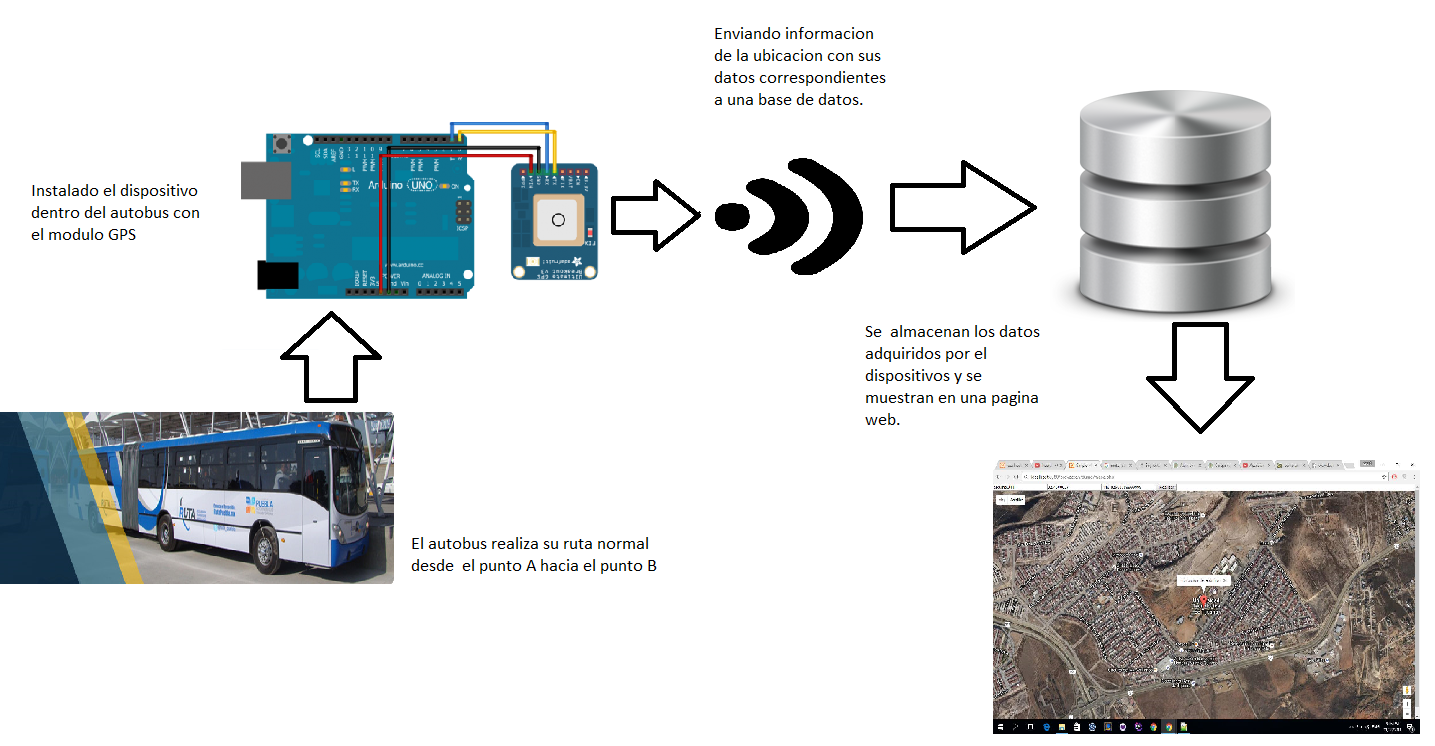
****

Ilustración Esquema del Proyecto

# **DIAGRAMA DEL PROYECTO**

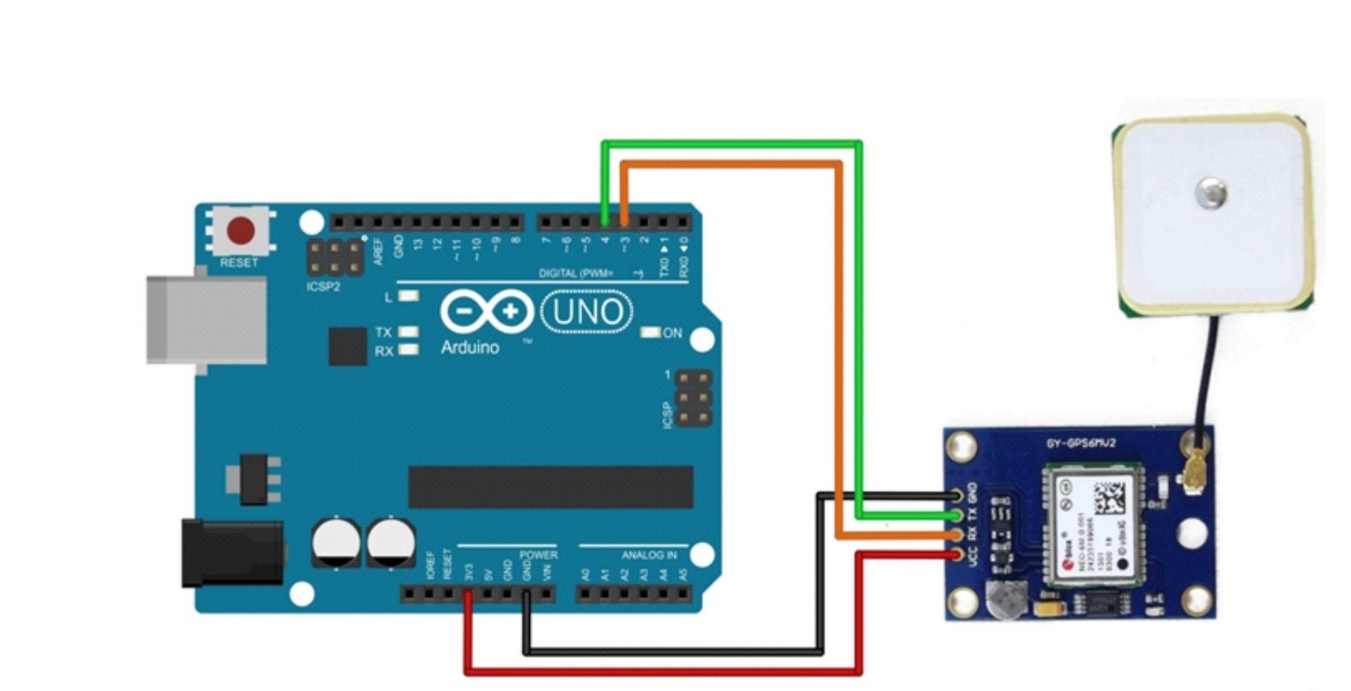


Ilustración Diagrama del Proyecto

# **RECURSOS Y MATERIALES**

**Hardware**

* Arduino UNO
* Ethernet Shield
* Modulo GPS

**Software**

* Servicio Apache
* Servicio MySQL
* Netbeans
* Arduino (Software para desarrollar)

# **Implementación del sensor**

El sensor que se implementará será el del GPS de la siguiente manera sobre la placa arduino(Sobre el ethernet shield).

* Conectando el pin 3.3V del Arduino UNO al pin Vcc del módulo GPS.
* Conectando el pin GND del Arduino UNO al pin GND del módulo GPS.
* Conectando el pin digital 4 del Arduino UNO al pin Tx del módulo GPS.
* Conectando el pin digital 3 del Arduino UNO al pin Rx del módulo GPS.

# **Programación del sensor con arduino**

#include <SPI.h>

#include <Ethernet.h>

#include <SoftwareSerial.h>

#include <TinyGPS.h>

// variables ethernet shield

byte mac[] = { 0x90, 0xA2, 0xDA, 0x0D, 0x9A, 0x47 };

IPAddress ip(192,168,1,12);

IPAddress serverIP(192,168,1,11);

EthernetClient client;

int serverPort=9000;

int msg;

TinyGPS gps;

SoftwareSerial ss(4, 3);

// variables sensor

float latitud;

float longitud;

String hora;

void setup() {

Serial.begin(115200);

ss.begin(9600);

//hora

Serial.print("Rutas Inteligentes"); Serial.println(TinyGPS::library\_version());

Serial.println("Monitoreo GPS");

Serial.println();

if (Ethernet.begin(mac) == 0) {

Serial.println("No se pudo configurar con dhpc, cambiando a ip statica");

Ethernet.begin(mac, ip);

}

Serial.println(Ethernet.localIP());

delay(1000);

Serial.println("Connectando al servidor ...");

if (client.connect(serverIP, serverPort)) {

Serial.println("Se pudo conectar con ip statica mensaje de prueba");

}

else {

Serial.println("Coneccion fallida");

}

}

void loop() {

bool newData = false;

unsigned long chars;

unsigned short sentences, failed;

// For one second we parse GPS data and report some key values

for (unsigned long start = millis(); millis() - start < 1000;)

{

while (ss.available())

{

char c = ss.read();

// Serial.write(c); // uncomment this line if you want to see the GPS data flowing

if (gps.encode(c)) // Did a new valid sentence come in?

newData = true;

}

}

if (newData)

{

float flat, flon;

unsigned long age;

gps.f\_get\_position(&flat, &flon, &age);

latitud = flat == TinyGPS::GPS\_INVALID\_F\_ANGLE ? 0.0 : flat, 6;

longitud = flon == TinyGPS::GPS\_INVALID\_F\_ANGLE ? 0.0 : flon, 6;

}

client.stop();

if(client.connect(serverIP, serverPort)){

String json = "{\"IdUnit\" :\"ABC-1234\", \"Locations\":{\"Latitude\" : \""+String(latitud)+"\", \"Longitude\": \""+String(longitud)+"\" }}";

Serial.println(json);

client.println(json);

client.flush();

}

else {

Serial.println("Coneccion fallida");

}

delay(5000);

}

# **Calibración del sensor**

Este sensor no necesita calibrarse puesto que viene calibrado desde fabrica, pero se tienen que tener unos puntos importantes para que este funcione correctamente y no haya errores cuando se esté implementando

* Se recomienda estar en un espacio a la intemperie sin obstáculos o techos que afecten o interfieran con el modulo, puesto que este tiene que realizar una conexión satelital, pero una vez que se conecta no importa si está bajo techo.
* Mantener alejado de tomas eléctricas.
* Mantener alejado de líquidos y/o alimentos.
* Tener el modulo dentro de un recipiente de preferencia transparente donde no le afecte el clima, debido a que es una tarjeta que no tiene protección alguna.
* No golpear o dañar el cable que conecta la antena.

# **Algoritmo sensor-Arduino**

# **Software**

# **Algoritmo de adquisición de datos Java-Arduino**

import Enumerators.UbicationResults;

import Exceptions.RecordNotFoundException;

import Models.Unit;

import Models.Location;

import java.net.\*;

import java.io.\*;

import org.json.JSONException;

import org.json.JSONObject;

import java.util.Date;

import java.text.DateFormat;

import java.text.SimpleDateFormat;

import Frames.FrameLocation;

public class Main {

static ServerSocket ss;

static Socket s;

static InputStreamReader isr;

static BufferedReader br;

static PrintStream ps;

public static void main(String[] args) {

final FrameLocation fm = new FrameLocation();

fm.show();

Thread socketThread = new Thread(new Runnable(){

@Override

public void run(){

try{

System.out.println("Openning Socket ...");//status

ss = new ServerSocket(9000); // socket port

System.out.println("Socket opened ...");// status again

//keep reading

boolean frame = true;

String idUnit;

while(true){

s = ss.accept(); //read data

isr = new InputStreamReader(s.getInputStream());

br = new BufferedReader(isr);// reader of buffer

String data = br.readLine();

DateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

try

{

JSONObject object = new JSONObject(data);

idUnit = object.getString("IdUnit");

JSONObject locations = object.getJSONObject("Locations");

String latitude = locations.getString("Latitude");

String longitude = locations.getString("Longitude");

Date date = new Date();

if(frame == true)

{

frame = false;

fm.textUnit.setText(idUnit);

fm.showLocationInfo();

}

fm.addRow(latitude, longitude, dateFormat.format(date));

try{

//timestamp

Unit u = new Unit(idUnit);

UbicationResults r = u.add\_ubication(latitude, longitude, dateFormat.format(date));

System.out.println(r.getMessage());

for(Location l : u.getLocations()){

System.out.println(l);

}

// System.out.println(data + "\n");//show data

}

catch(RecordNotFoundException ex){

System.out.println(ex.getMessage());}

}

catch(JSONException ex){

System.out.println(ex);

}

//System.out.println(data + "\n");//show data

//send a knowledge to client

if (data != null){

ps = new PrintStream(s.getOutputStream());

ps.println("Data received ...");

//if(idUnit == "")

//{

// fm.textUnit.setText(idUnit);

//}

//fm.showLocationInfo();

}

}

}

catch(IOException ex)

{

System.out.println(ex.getMessage());

}

}

});

//start socket

socketThread.start();

}

}

# **Diagrama de flujo**

# **Base de datos modelo E-R**

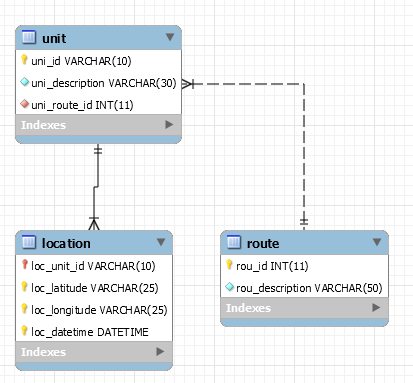


Ilustración Modelo Entidad-Relación

# **Almacenamiento de datos Java-Arduino**

package Models;

import MySql.MySqlConnection;

import java.util.Date;

import MySql.MySqlConnection;

import Exceptions.RecordNotFoundException;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.Date;

import Enumerators.UbicationResults;

import java.sql.CallableStatement;

public class Unit {

private static PreparedStatement command;

private static ResultSet result;

CallableStatement procedure;

private String id;

private String descritpion;

private String route;

private Location locations;

public String getId() {

return id;

}

public String getDescritpion() {

return descritpion;

}

public String getRoute() {

return route;

}

public Unit(String id) throws RecordNotFoundException{

try{

String query = "SELECT uni\_id, uni\_description, rou\_description FROM unit, route WHERE uni\_route\_id = rou\_id AND uni\_id = ?";

command = MySqlConnection.getConnection().prepareStatement(query);

command.setString(1,id);

result = command.executeQuery();

result.first();

if(result.getRow() > 0){

this.id = result.getString("uni\_id");

this.descritpion = result.getString("uni\_description");

this.route = result.getString("rou\_description");

}

else{

throw new RecordNotFoundException(this.getClass().getName(),String.valueOf(id));

}

}

catch(SQLException ex){

}

}

public UbicationResults add\_ubication(String longitud, String latitud,String nowDate){

int result = 0;

String call = "call sp\_new\_location(?,?,?,?,?)";

///(acount\_id,concept\_id,ammount,out result)

try{

procedure = MySqlConnection.getConnection().prepareCall(call);

//parameters

procedure.setString(1,this.id);

procedure.setString(2,latitud);

procedure.setString(3,longitud);

procedure.setString(4,nowDate);

//out parameters

procedure.registerOutParameter(5,java.sql.Types.INTEGER);

//ejecute

procedure.executeUpdate();

//recived out param

result = procedure.getInt(5);

//uptdate balance attribute

}

catch(SQLException ex){

}

return UbicationResults.values()[result];

}

public ArrayList<Location> getLocations(){

ArrayList<Location> list = new ArrayList<Location>();

String query = "SELECT loc\_id, loc\_datetime, loc\_latitude, loc\_longitude"

+ "FROM unit join locations on loc\_unit\_id = uni\_id where uni\_id = ?";

try{

command = MySqlConnection.getConnection().prepareStatement(query);

command.setString(1,this.id);

result = command.executeQuery();

while(result.next()){

int locId = result.getInt("loc\_id");

Date date = result.getDate("loc\_datetime");

String longitude = result.getString("loc\_longitude");

String latitude = result.getString("loc\_latitude");

list.add(new Location(locId,longitude,latitude,date));

}

}

catch(SQLException ex){

}

return list;

}

}

# **Dashboard de grafica**

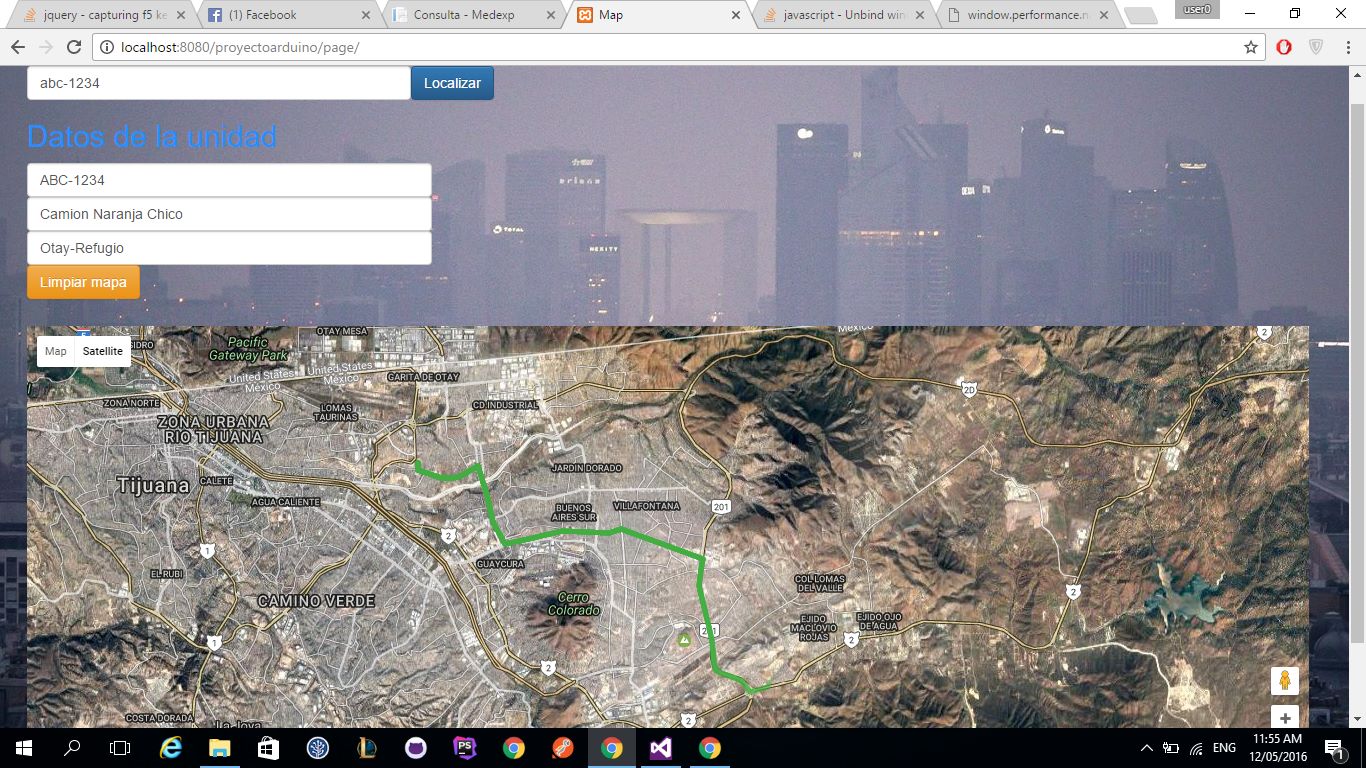


Ilustración Dashboard

# **Resultados Obtenidos**

Los resultados obtenidos fue que se obtuvo la latitud y longitud mediante el modulo GPS que se implementó en la placa Arduino, que mediante la conexión por sockets se logró realizar él envió correcto de los datos hacia la base de datos, en donde almacenábamos todos los valores devueltos, así como la hora precisa, donde la misma estaba enlazada a una aplicación web en donde se mostraba la ruta especifica de los datos adquiridos en la que se podía visualizar el recorrido que el modulo GPS y el Arduino realizaron durante un periodo de tiempo, dentro de un mapa con la implementación de tecnologías de Google Maps.

En este caso para el proyecto se realizó una ruta de prueba la cual no es oficial en nuestra ciudad, por motivos de incompatibilidad de tecnologías y problemas técnicos entre materias pero una vez que este proyecto esté en otro cuatrimestre y se puedan utilizar las tecnologías correspondientes y se llegue a la meta que es el 100% , se implementará en los autobuses de Tijuana para beneficiar a los ciudadanos y ayudar a mejorar el transporte público.

Por lo que se puede decir que la fase del proyecto de este cuatrimestre fue concluida de manera satisfactoria, debido a que se logró concretar todos y cada uno los objetivos planteados inicialmente en este proyecto. De manera que sigamos progresando en la ingeniería, se ira agregando más innovaciones a este proyecto, pero por el momento todo salió acuerdo a lo estipulado.

# **Pruebas**

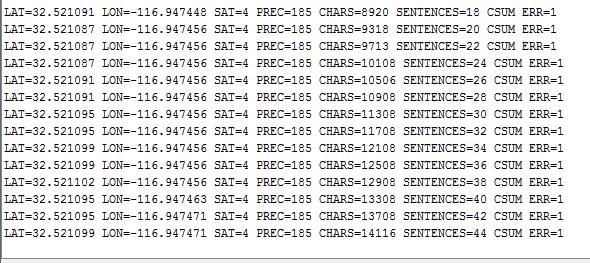


Ilustración Prueba: Obtención de datos vía modulo GPS

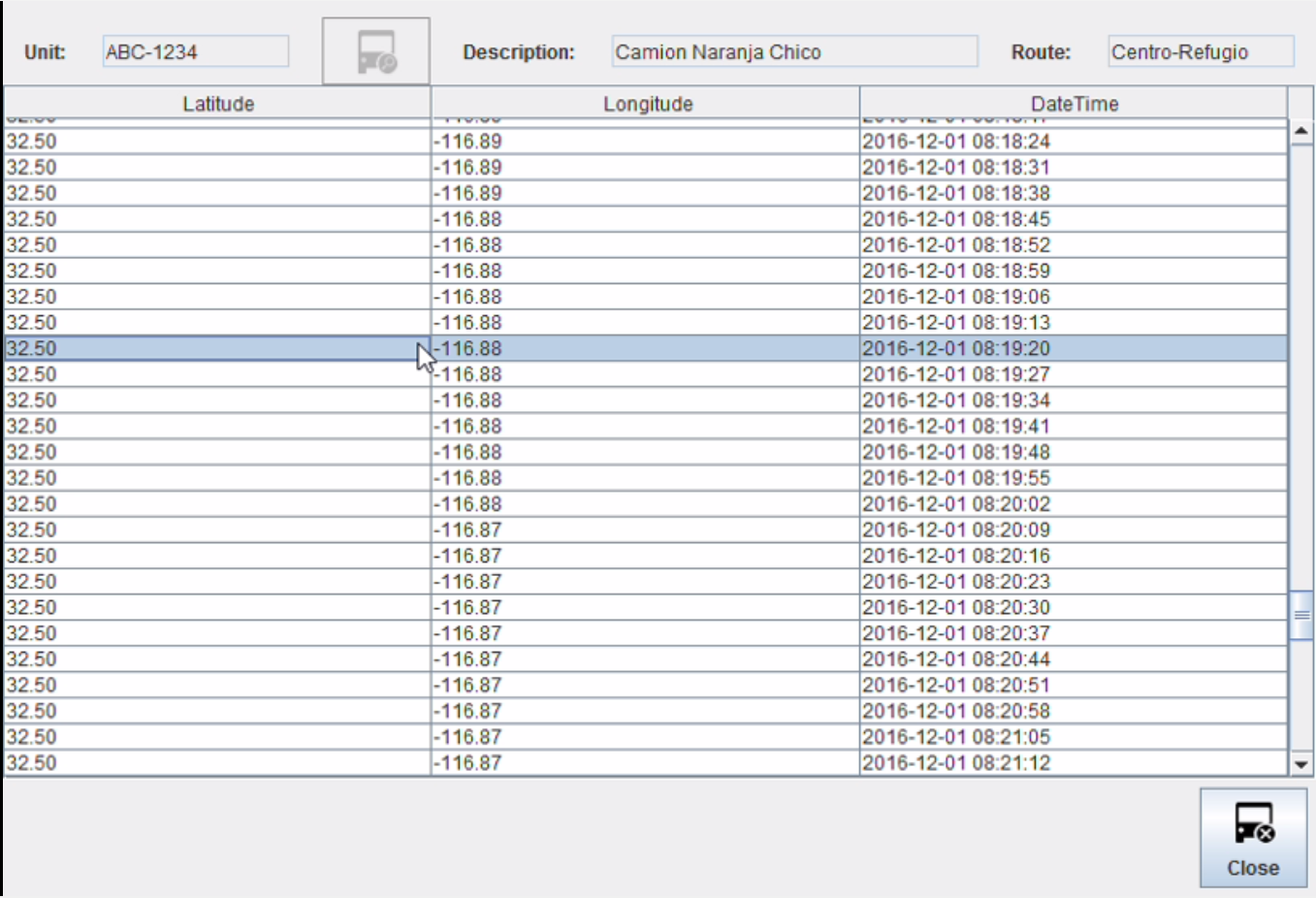


Ilustración Prueba: Envió de datos desde arduino hacia Java