



**TECNOLÓGICO NACIONAL DE MÉXICO**  
**CAMPUS TUXTLA GUTIÉRREZ**

**PROYECTO DE RESIDENCIA PROFESIONAL**

“Sistema para la medición de hidrocarburos mediante sensores basado en  
Android”

**Ingeniería en Sistemas Computacionales**

**ALUMNOS**

Jorge Hiram Llaven Velázquez	14270095
Francisco Javier López Juárez	14270101

**ASESOR INTERNO**

M.C. Octavio Ariosto Ríos Tercero

Tuxtla Gutiérrez, Chiapas, Enero - Junio 2018



## **Agradecimientos**

A Dios, por darme la vida y la salud.

A mis Padres y Hermanos, por su apoyo incondicional.

A los docentes del instituto tecnológico de Tuxtla Gutiérrez, quienes me han brindado su amistad y han compartido gran parte de sus conocimientos.

A todas las amistades que he conseguido a lo largo de la vida, quienes me han apoyado en todo momento para seguir adelante.

## Resumen

El presente proyecto está desarrollado de acuerdo a las necesidades que presenta el instituto tecnológico de Tuxtla Gutiérrez; quienes han percibido que la contaminación ambiental que rodea a la urbe se ha venido incrementado desde hace años atrás.

El principal objetivo de esta investigación fue realizar un estudio para la implementación de un Sistema de Monitoreo de Inmisión de gases contaminantes como el benceno, el tolueno y el xileno, medido en partes por millón (ppm); gases que afectan a la ciudad y a sus habitantes; y así, tener información en tiempo real de los niveles a los cuales está expuesta la población. Además, el sistema de monitoreo mide temperatura, en grados centígrados (°C) y humedad en porcentaje (%).

Las principales industrias con las que cuenta la ciudad de Tuxtla Gutierrez tienen instalados sistemas de monitoreo de emisión de gases contaminantes que son utilizados para el mejoramiento de sus procesos industriales, así como también para controlar y disminuir la emisión de contaminantes a la atmósfera. Sin embargo, estos datos no están a disposición de la población. Por lo tanto, para facilitar a la población que realice consultas sobre los niveles de contaminación de los gases que mayor efecto tienen sobre la salud humana se ofrece una herramienta que consiste en un Sistema de Monitoreo de Gases Contaminantes que podrá ser visualizado en nuestra app.

Es difícil eliminar por completo la emisión de contaminantes, pero si es posible reducirla pese a la aparición de otras industrias y el crecimiento del parque automotor que hacen que existan gases contaminantes en el ambiente.

# Contenido

<b>Agradecimientos .....</b>	<b>3</b>
<b>Resumen .....</b>	<b>4</b>
<b>Contenido.....</b>	<b>5</b>
<b>1. Introducción .....</b>	<b>7</b>
<b>2. Descripción de la Empresa .....</b>	<b>9</b>
<b>2.1 Misión.....</b>	<b>9</b>
<b>2.2 Visión .....</b>	<b>9</b>
<b>2.3 Valores .....</b>	<b>9</b>
<b>2.4 Organigrama.....</b>	<b>10</b>
<b>3. Problemas a resolver.....</b>	<b>11</b>
<b>4. Objetivos .....</b>	<b>12</b>
<b>4.1 Objetivo general .....</b>	<b>12</b>
<b>4.2 Objetivos específicos .....</b>	<b>12</b>
<b>5. Justificación.....</b>	<b>13</b>
<b>6. Marco Teórico .....</b>	<b>14</b>
<b>6.1 Marco Referencial .....</b>	<b>14</b>
<b>6.1.1 Contaminación .....</b>	<b>14</b>
<b>6.1.2 Benceno.....</b>	<b>16</b>
<b>6.1.3 Tolueno.....</b>	<b>17</b>
<b>6.1.4 Xileno.....</b>	<b>19</b>
<b>6.1.5 Normas y reglamentos en México .....</b>	<b>20</b>
<b>6.2 Marco Computacional.....</b>	<b>22</b>
<b>6.2.1 Arduino .....</b>	<b>22</b>
<b>6.2.2 Servicios web.....</b>	<b>25</b>
<b>6.2.3 PHP .....</b>	<b>29</b>
<b>6.2.4 MYSQL.....</b>	<b>29</b>
<b>6.2.5 MpAndroidChart .....</b>	<b>30</b>
<b>7. Procedimiento de las Actividades Realizadas .....</b>	<b>32</b>

7.1	Análisis del diseño del prototipo.....	32
7.2	Calibración de los sensores.....	33
7.3	Programación de módulo .....	34
7.4	Desarrolló de los servicios web.....	34
8.	Resultados obtenidos .....	36
8.1	Fase de Análisis .....	36
8.1.1	Requisitos Funcionales de la aplicación móvil.....	36
8.1.2	Descripción de actor(es).....	36
8.1.3	Diagrama de casos de uso.....	37
8.1.4	Plantillas de casos de uso .....	37
8.1.5	Requisitos no funcionales .....	39
8.2	Fase de diseño .....	40
8.2.1	Esquema conceptual del sistema .....	40
8.2.2	Diagrama de clases de la aplicación.....	41
8.2.3	Diseño detallado de la aplicación móvil .....	42
8.2.4	Programación de la placa Arduino.....	46
8.2.5	Programación de los servicios web.....	47
9.	Resultados finales .....	51
9.1	Diseño de Contenedor de Acrílico.....	51
9.2	Diseño del prototipo .....	52
9.2.1	Esquema del prototipo.....	52
9.3	Vistas de la aplicación y del prototipo .....	53
9.3.1	Vista de colocación del prototipo en contenedor acrílico .....	53
9.3.2	Vistas de la aplicación Android.....	56
10.	Conclusión.....	60
11.	Competencias desarrolladas.....	61
11.1	Competencias genéricas.....	61
11.2	Competencias desarrolladas en la elaboración de la aplicación móvil .....	61
11.3	Competencias desarrolladas en la elaboración del prototipo basado en Arduino .....	61
12.	Fuentes de Información .....	62

## 1. Introducción

El proyecto “Bromelia sp. como bioindicador de BTX y cambio climático” se realizó en las instalaciones del instituto tecnológico de Tuxtla Gutiérrez la cual es una institución educativa de alto reconocimiento y de nivel superior, especializada en la enseñanza y aprendizaje de tecnologías.

Este proyecto pertenece al área de posgrado del nivel de Maestría en Ciencias en Ingeniería Bioquímica (MCIBQ) del ITTG el cual es uno de los posgrados más importantes del estado de Chiapas y se encuentra dentro del Padrón Nacional de Programas de Calidad (PNPC) del CONACYT.

Actualmente, la LGEEPA cuenta con los reglamentos en materias de Impacto ambiental (nuevo reglamento publicado en el D.O.F. del 30 de mayo de 2000), Residuos peligrosos (publicado en el D.O.F. el 25 de noviembre de 1988), Contaminación originada por ruido (publicado en el D.O.F. 6 de diciembre de 1982), Prevención y control de la contaminación atmosférica (publicada en el D.O.F. el 25 de noviembre de 1988),

Prevención y control de la contaminación generada por los vehículos automotores que circulan por el Distrito Federal y los municipios de su zona conurbada (publicado en el D.O.F. el 25 de noviembre 1998), Para el transporte terrestre de materiales y residuos peligrosos (Publicado en el D.O.F. el 7 de abril de 1993), Áreas naturales protegidas (nuevo reglamento publicado en el D.O.F. el 30 de noviembre de 2000) y Auditoría ambiental (nuevo reglamento publicado en el D.O.F. el 29 de noviembre de 2000). Esta ley coloca en los escenarios a la SEMARNAT, PROFEPA e INECC. Las dependencias gubernamentales del gobierno federal INECC y PROFEPA, son las encargadas de la gestión y vigilancia en materia ambiental, corresponde a la primera diseñar, conducir y evaluar la política nacional en materia de ecología y protección al ambiente, la segunda se encarga de la verificación del cumplimiento de las normas y legislación en materia ambiental, mientras que la PROFEPA se encarga de vigilar que se cumplan los términos aprobados (PROFEPA – SEDESOL, 1993). La contra parte estatal encargada de hacer cumplir las regulaciones ambientales coloca en los escenarios a la Secretaría de Medio Ambiente e Historia Natural (SEMAHN) y al Instituto de Protección al Medio Ambiente del municipio de Tuxtla Gutiérrez. (Enciso Sáenz, 2013) En atención a las directrices arriba expuestas se realizó el Estudio de emisiones y actividad vehicular en la ciudad de

Tuxtla Gutiérrez, Chiapas, con la finalidad de obtener información que permitiera elaborar un diagnóstico del desempeño ambiental, la actividad y las características de los vehículos automotores que circulan en esta localidad. En el informe se afirma que parque vivo o en circulación al año 2011 era de 190,446 vehículos, lo que significó 253 vehículos por 1000 habitantes y el análisis estadístico de los resultados del estudio indicó que la media de las emisiones de monóxido de Carbono fue de 0.99 (% en vol.) y la de hidrocarburos 543(ppm) (INECC - SEMARNAT, 2012). Desde esta perspectiva, es evidente que Tuxtla Gutiérrez, Chiapas carece de estudios científicos que describen sus condiciones meteorológicas y de dispersión de las emisiones atmosféricas de las fuentes móviles, tampoco hay información de los precursores de GEI ni de sus efectos en los ecosistemas.

Por lo anterior, el propósito de esta investigación se enfoca en la detección y medición de los gases de hidrocarburos mediante un prototipo que logre graficar los volúmenes de estos y pueda ser utilizado en diferentes ambientes donde la información pueda ser vista desde diferentes dispositivos en cualquier lugar que tenga una conexión a internet.



## 2. Descripción de la Empresa

El TECNM campus Tuxtla Gutiérrez; es una universidad pública de tecnología, ubicada en la Ciudad de Tuxtla Gutiérrez, Chiapas, México. Es una Institución educativa pública de educación superior, que forma parte del Sistema Nacional de Institutos Tecnológicos de México. El Instituto también está afiliado a la Asociación Nacional de Universidades e Instituciones de Educación Superior (ANUIES), zona Sursureste.

Fue fundado el 22 de octubre de 1972, por el entonces Gobernador del Estado, Dr. Manuel Velasco Suárez, inicialmente con el nombre de Instituto Tecnológico Regional de Tuxtla Gutiérrez (ITRTG), posteriormente se llamaría el Instituto Tecnológico de Tuxtla Gutiérrez (ITTG).

Su lema es Ciencia y Tecnología con Sentido Humano y su actual director es el M.E.H José Luis Méndez Navarro.

### 2.1 Misión

“Formar de manera integral profesionales de excelencia en el campo de la ciencia y la tecnología con actitud emprendedora, respeto al medio ambiente y apego a los valores éticos.”

### 2.2 Visión

Ser una Institución de excelencia en la educación superior tecnológica del sureste, comprometida con el desarrollo socioeconómico sustentable de la región.

### 2.3 Valores

- El ser humano
- El espíritu de servicio.
- El liderazgo.
- El trabajo en equipo.
- La calidad.
- El alto desempeño.
- Respeto al medio ambiente.

La ubicación de la empresa se encuentra en KM 29020, Carr. Panamericana 1080, Tuxtla Gutiérrez, Chiapas.<sup>1</sup>

---

<sup>1</sup> [https://es.wikipedia.org/wiki/Instituto\\_Tecnologico\\_de\\_Tuxtla\\_Gutierrez](https://es.wikipedia.org/wiki/Instituto_Tecnologico_de_Tuxtla_Gutierrez)

## 2.4 Organigrama

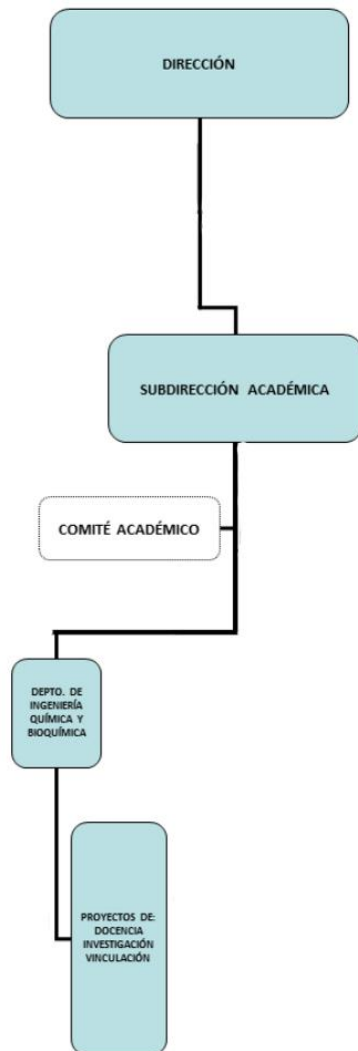


Figura 1: Organigrama del departamento de Bioquímica

Ingeniería Química es la rama de la Ingeniería que se dedica al estudio, síntesis, desarrollo, diseño, operación y optimización de todos aquellos procesos industriales que producen cambios físicos, químicos y/o bioquímicos en los materiales. Ingeniería Química es la profesión en la cual el conocimiento de la matemática, química y otras ciencias básicas, ganados por el estudio, la experiencia y la práctica, es aplicado con juicio para desarrollar maneras económicas de usar materiales y energía para el beneficio de la humanidad.

### **3. Problemas a resolver**

- Calibrar los sensores para que midan correctamente los hidrocarburos (Benceno, Tolueno y Xileno).
- Que los datos que cense el Arduino sean recibidos y visualizados por la app móvil por medio de una conexión inalámbrica.
- Representar en la aplicación por medio de una gráfica la cantidad de los hidrocarburos medidos.
- Tener un registro de la cantidad de los hidrocarburos que se van a censar.

## **4. Objetivos**

### **4.1 Objetivo general**

Desarrollar una aplicación móvil (app) para la visualización, graficación y cuantificación de los datos con el uso de sensores de hidrocarburos (Benceno, Tolueno, Xileno) que serán obtenidos mediante una conexión inalámbrica en un dispositivo Arduino.

### **4.2 Objetivos específicos**

- Medir los niveles de concentración de los gases.
- Visualizar gráficos de concentración en la app.

## **5. Justificación**

Tuxtla Gutiérrez, carece de estudios científicos que describan las condiciones de las emisiones atmosféricas, precursores de gases de efecto invernadero (GEI), ni de sus efectos en los ecosistemas y el ambiente. El monitoreo convencional del ambiente urbano e industrial requiere de técnicas cada vez más costosas y complejas, por lo cual se hace necesario el uso de mecanismos alternativos que permitan obtener información a precios más accesibles. Esta aplicación en conjunto con el prototipo desarrollado en Arduino permitirá al usuario obtener esos datos de manera más simple y sencilla.

## **6. Marco Teórico**

### **6.1 Marco Referencial**

#### **6.1.1 Contaminación**

##### ***6.1.1.1 Definición de contaminación***

Se conoce como contaminación, a la transmisión y difusión de sólidos, líquidos o gases tóxicos a medios como la atmósfera, el agua, o el suelo. De acuerdo al origen, existe contaminación proveniente de eventos naturales como: erupciones volcánicas, incendios forestales, entre otros, o de los desechos de las actividades del ser humano como:

Procesos industriales, combustión de hidrocarburos, incremento de automóviles en las ciudades, ensayos nucleares, desechos orgánicos, por citar algunos casos; y que estos, son causa de múltiples problemas a la salud de las personas y a los bienes en general.

La contaminación es difícil de eliminar, pero si se realiza un control y se toma conciencia de los problemas que causa, es posible disminuir sus niveles de contaminación para así preservar la salud de todos los seres vivos en general y especialmente la de la especie humana.

##### ***6.1.1.2 Tipos de contaminación:***

De acuerdo al recurso natural que afecta la contaminación puede ser:

- a) Contaminación atmosférica (aire) Este tipo de contaminación se da por la liberación de sustancias químicas y partículas en la atmósfera alterando su composición, lo que ocasiona un riesgo para la salud de las personas y de otros seres vivos. Los gases contaminantes del aire más comunes son: el monóxido de carbono, el dióxido de azufre y los óxidos de nitrógeno.

algunos ejemplos de los gases orgánicos volátiles serían el benceno, tolueno y xileno producidos en la combustión en las industrias y de los motores de los vehículos. La contaminación atmosférica se dice que es local, cuando los

efectos ligados al foco de emisión afectan solo a las inmediaciones del mismo, o es global, cuando las características del contaminante afectan al equilibrio del planeta y zonas distantes a los focos emisores, entre estos se pueden citar la lluvia ácida y el calentamiento global.

- b) Contaminación hídrica (agua de mares, ríos, lagos) Este tipo de contaminación se da por la liberación de residuos y contaminantes que son drenados a las alcantarillas y luego llegan hacia los ríos, penetrando en aguas subterráneas o descargando directamente en lagos y mares; otros que contaminan los océanos y playas.

Son los desechos marinos que en su mayoría son plásticos, algunas veces se acumulan en alta mar como en la gran mancha de basura del Pacífico norte; los derrames de petróleo en los pozos petroleros o por fugas en sus tuberías de transporte, también son contaminantes de los efluentes hídricos.

- c) Contaminación edafológica (suelo) Este tipo de contaminación se da cuando productos químicos son liberados por un derrame o filtraciones sobre la tierra. Entre los contaminantes del suelo están los hidrocarburos como el petróleo y sus derivados.

los metales pesados presentes en las baterías, los herbicidas y plaguicidas rociados en los monocultivos producidos por la industria; también los vertederos y cinturones ecológicos que entierran grandes cantidades de basura en las ciudades.

## 6.1.2 Benceno

El benceno es un hidrocarburo aromático de fórmula molecular  $C_6H_6$ , (originariamente a él y sus derivados se le denominaban compuestos aromáticos debido a la forma característica que poseen) también es conocido como benzol. En el benceno cada átomo de carbono ocupa el vértice de un hexágono regular, aparentemente tres de las cuatro valencias de los átomos de carbono se utilizan para unir átomos de carbono contiguos entre sí, y la cuarta valencia con un átomo de hidrógeno.

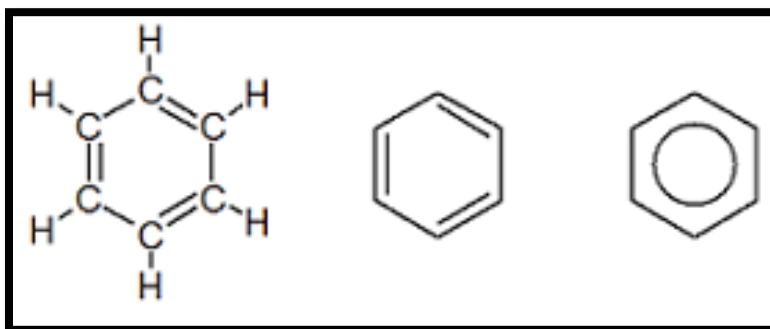


Figura 2: Benceno

### 6.1.2.1 Toxicocinética del Benceno

El metabolismo del benceno empieza con una oxidación a epóxido catalizado principalmente por la enzima. El óxido de benceno se metaboliza por tres diferentes rutas 1) Conjugación con GHS para formar ácido premercapturico, el cual se convierte en ácido fenil mercapturico; 2) Arreglo no enzimático para formar fenol y 3) Hidración por la enzima epóxido hidratasa a benceno dehidrodiol el cual puede ser oxigenado por la enzima dihidrodiol dehidrogenasa a catecol. Si el fenol se hidroliza en su posición orto, se puede producir más catecol.

El catecol puede convertirse a p-benzoquinona. Se cree que él o- y p-benzoquinonas son los últimos metabolitos tóxicos del benceno. Pero también puede formarse otros metabolitos tóxicos, el muconaldehído a partir de los anillos abiertos del óxido de benceno. El muconaldehído sufre una serie de reacciones que llevan al ácido t, t-mucónico, el cual puede encontrarse en orina.



### 6.1.2.2 Análisis de Benceno en muestras biológicas y ambientales

Se han desarrollado métodos analíticos para medir benceno en aire exhalado, sangre, y en varios tejidos. Estos análisis se realizan principalmente por cromatografía de gases (CG) acoplado a detector de ionización de flama (DIF), detector de fotoionización (DFI) o de masas (SM).

Uno de los métodos más sensibles (3 ppt, partes por trillón) es la cromatografía de gases de alta resolución acoplado a un detector de masas (CGAR/SM) para detectar benceno en aire exhalado y en sangre. Para prevenir la contaminación de las muestras se siguen rigurosos métodos para su recolección y su preparación.

### 6.1.3 Tolueno

También conocido con el nombre de metilbenceno, sigue la fórmula  $C_6H_5CH_3$ . El tolueno se usa como materia prima, siendo ésta bastante importante pues a partir de ella se consigue obtener importantes compuestos, como por ejemplo, los derivados bencénicos, ácido benzoico, fenoles, caprolactama, sacarinas, y el conocido diisocianato de tolueno (TDI) a partir del cual se fabrica poliuretano, diferentes colorantes y perfumes, medicinas, detergentes y el TNT entre otras cosas.

Podemos encontrarlo en su forma natural, formando parte del petróleo crudo, además de en el mencionado árbol del Tolú. Pero se puede sintetizar y producirlo a partir de la manufacturación de la gasolina, así como también de otros tipos de combustibles partiendo del petróleo, o también en la producción del coque partiendo del carbón.



Figura 3: Tolueno

### **6.1.3.1 Toxicocinética del Tolueno**

El Tolueno inhalado se adsorbe en pulmones y se transporta por la sangre a otros tejidos y órganos, es absorbido en el tracto digestivo, se acumula rápidamente en el cerebro. Subsecuentemente se acumula en tejido adiposo en donde se retiene en altos niveles de concentración.

El metabolismo del Tolueno se realiza vía citocromos P450s transformándolo en alcohol bencénico y en pequeñas cantidades de cresoles. El alcohol bencénico se convierte a su vez en ácido bencénico por la acción de la enzima alcohol deshidrogenasa, el cual se conjuga y se elimina por orina.

Al igual que el Benceno el Tolueno se comporta como depresor de SNC, son varios los órganos afectados por este solvente, los índices más altos de afectación se presentan en hígado, riñón y órganos del SNC y del sistema nervioso periférico. Es causante de hematopatía, tubulopatía y distal, ataxia, temblores y lateraciones en el comportamiento, así como de polineuropatías.

### **6.1.3.2 Análisis de Tolueno en muestras biológicas y ambientales**

La mayoría de los métodos para el análisis de Tolueno de fluidos biológicos (sangre, leche materna, orina, aliento, tejido cerebral y adiposo) usan técnicas de cromatografía de gases.

El aliento usualmente se colecta en trampas adsorbentes o en bolsa para muestra o contenedores y se analizan por CG. El método de análisis con mayor sensibilidad (0.088µg/l) es la cromatografía de gases acoplado a detector de masas (CG/SM).

## 6.1.4 Xileno

El Dimetil benceno es usado como un solvente para gomas, aceites y resinas epóxicas y otras resinas como un intermediario en la manufactura de químicos orgánicos, usados en la manufactura de plásticos y textiles sintéticos. También es usado en la manufactura de pinturas, resinas, insecticidas, peróxido de hidrógeno y osciladores de cristal de cuarzo.

También se usa como un agente para el combustible de alto octanaje de motores y aviones. Es un buen desengrasante.

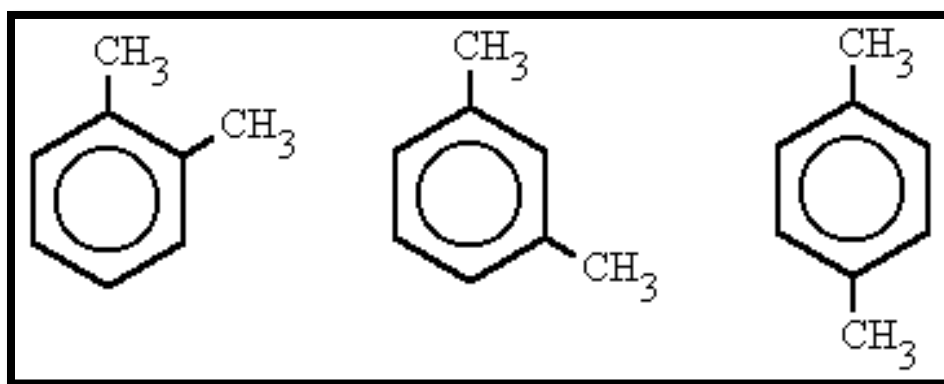


Figura 4: Xileno

### 6.1.4.1 Toxicocinética de los Xilenos

La toxicidad de los Xilenos es muy parecida a la que producen otros disolventes aromáticos como el Benceno. El Xileno es adsorbido por pulmones y el tracto intestinal, y es distribuido a los tejidos de acuerdo a su irrigación sanguínea o contenido de lípidos. El metabolismo de los tres isómeros del Xileno, es una oxidación del grupo metil y la subsecuente conjugación con glicina (ácido metil hipúrico).

La hidroxilación del anillo y la conjugación con el ácido glucorónico, que es aproximadamente del 2%. El Xileno sin transformación puede ser eliminado por exhalación y los metabolitos de su transformación son arrojados por la orina. La mayoría de la gente puede empezar a detectar el olor del Xileno en el aire cuando está en concentraciones de 0.08 a 3.7 ppm y de 0.53 a 1.1 ppm en agua.

#### **6.1.4.2 Análisis de Xileno en muestras biológicas y ambientales.**

Existe un número limitado de métodos utilizados para determinar las concentraciones traza de Xileno en tejidos (pescado y ratón) y fluidos biológicos (sangre humana, orina y aliento exhalado), éstos incluyen la cromatografía de gases acoplado a espectrometría de masas (CG/SM) que es el más sensible para detectarlo en sangre ( de 1ng/L de m-xileno), espectrometría de gases acoplado a un detector de ionización de flama con nitrógeno (CG/DIF) y la cromatografía líquida de alta resolución (CLAR).

La cromatografía de gases equipado con el detector adecuado es el instrumento analítico usado para la determinación de Xileno en muestras ambientales (aire, agua subterránea, agua superficial, suelo, sedimento y residuos). El equipo con el mayor nivel de detección es el equipado con un detector de captura de electrones (1ng/L de Xileno).

Para evitar las fugas del Xileno en el aire se debe realizar con precaución su aislamiento, recolección y almacenamiento. Las normas y los reglamentos, que regulan los límites de exposición a Xileno están reportados por agencias internacionales como ACGIH (TLV-TWA = 435 mg/m<sup>3</sup> y Tel = 655 mg/m<sup>3</sup>), NIOSH (REL = 435 mg/m<sup>3</sup> y STEL= 655 mg/m<sup>3</sup>), etc.

#### **6.1.5 Normas y reglamentos en México**

Las normas y reglamentos con los que actualmente cuenta México para determinar los niveles de concentración del Benceno, Tolueno y Xilenos en muestras ambientales y biológicas, se presentan a continuación:

1. Norma Oficial Mexicana NOM-047-SSA1-1993, Que establece los límites biológicos máximos permisibles de disolventes orgánicos en el personal ocupacionalmente expuesto.
2. NOM-127-SSA1-1994, Salud ambiental, agua para uso y consumo humano- límites permisibles de calidad y tratamientos a que debe someterse el agua para su potabilización.
3. CCE-CCA-01-1989. Criterios ecológicos de calidad del agua CC-CCA-001-1989.

4. NMX-AA-141-SCFI-2002 Suelos- benceno, tolueno, etilbenceno y xilenos (BTEXs) por cromatografía de gases con detectores de espectrometría de masas y fotoionización. Método de prueba.

5. Norma Oficial Mexicana NOM-138-SEMARNAT/SS-2003 Límites máximos permisibles de hidrocarburos en suelo y las especificaciones para su caracterización y remediación.

Se observa una baja vigilancia de la concentración de los BTX en alimentos, De acuerdo al listado, en lo que respecta a alimentos contaminados con estos disolventes hace falta más trabajo de investigación a nivel nacional, para la adecuación de metodologías analíticas que permitan evidenciar su presencia en estas matrices, así como para fomentar la creación de nuevas leyes y reglamentos al respecto.

## 6.2 Marco Computacional

### 6.2.1 Arduino

#### 6.2.1.1 ¿Qué es Arduino?

Cuando se habla de Arduino, es necesario tener en cuenta que este se halla constituido por tres elementos fundamentales y que todos ellos se denominan bajo en mismo nombre (Arduino), estos componentes son: (Profeyoni, 2015)

- El hardware o placas de Arduino
- El entorno de programación o IDE
- El lenguaje de programación.

En general estos componentes fueron creados con la finalidad de tener una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles y fáciles de usar. Se creó para artistas, diseñadores, aficionados y cualquiera interesado en crear entornos u objetos interactivos.

#### 6.2.1.2 Hardware

Actualmente se usa un único microcontrolador que se encarga de llevar a cabo la comunicación y sobre el que también se descargan las instrucciones a ejecutar, tiene una serie de aditamentos llamados Shell (escudos o cascaras) los cuales les dan más funcionalidad a las placas. (Profeyoni, 2015)



Figura 5: Estructura de la placa Arduino uno. Tomada de (Profeyoni,2015)

1. Conector USB, que puede ser tipo B o mini, este provee la comunicación para la programación y la toma de datos, también provee una fuente de 5VDC para alimentar al Arduino.
2. Regulador de voltaje de 5V, se encarga de convertir el voltaje ingresado por el plug 3, en un voltaje de 5V regulado. necesario para el funcionamiento de la placa y para alimentar circuitos externos.
3. Plug de conexión para fuente de alimentación externa, el voltaje que se suministra por aquí debe ser directo y estar entre 6V y 18V, incluso 20V.
4. Puerto de conexiones; constituido por 6 pines de conexión con las siguientes funciones: RESET, permite resetear el microcontrolador al enviarle un cero lógico. Pin 3.3V, este pin provee una fuente de 3.3VDC para conectar dispositivos externos. Pin 5V, es una fuente de 5VDC para conectar dispositivos externos. Dos pines GND, que proveen la salida de cero voltios para dispositivos externos. Pin Vin, este pin está conectado con el positivo del plug 3 por lo que se usa para conectar la alimentación de la placa con una fuente externa de entre 6 y 12VDC en lugar del plug 3 o la alimentación por el puerto USB. Este puerto esta modificado en la versión R3 de Arduino Uno.
5. Puerto de entradas análogas, aquí se conectan las salidas de los sensores análogos. Estos pines solo funcionan como entradas recibiendo voltajes entre cero y cinco voltios directos.
6. Microcontrolador Atmega 328, es el microcontrolador implementado en los Arduino uno.
7. Botón de RESET, este botón, así como el pin mencionado anteriormente permiten resetear el microcontrolador haciendo que reinicie el programa. En la versión R3 este pulsador se ubica arriba del conector USB, esto es un acierto pues al colocarle las Shield encima del Arduino, se perdía la opción de resetear dado que este pulsador quedaba tapado.
8. Pines de programación ICSP, son usados para programar microcontroladores en protoboard o sobre circuitos impresos sin tener que retirarlos de su sitio.
9. Led On, enciende cuando el Arduino esta encendido.
10. Leds de recepción y transmisión, estos se encienden cuando la tarjeta se comunica con el PC. El Tx indica transmisión de datos y el Rx recepción.
11. Puerto de conexiones, está constituido por los pines de entradas o salidas digitales desde la cero hasta la 7. La configuración como entrada o salida debe ser incluida en el programa. Cuando se usa la terminal serial es conveniente no utilizar los pines cero (Rx) y uno (Tx). Los pines 3, 5 y 6 están precedidos por el símbolo ~, lo que indica que permiten su uso como salidas controladas por ancho de pulso PWM.
12. Puerto de conexiones, incluye 5 entradas o salidas adicionales (de la 8 a la 12), las salidas 9, 10 y 11 permiten control por ancho de pulso; la salida 13 es un poco diferente pues tiene conectada una resistencia en serie, lo que permite conectar un led directamente entre ella y tierra. Finalmente hay una salida a tierra GND y un pin AREF que permite ser empleado como referencia para las entradas análogas.
13. Este led indica el estado del pin 13.

14. Pines de programación ICSP, son usados para programar microcontroladores en protoboard o sobre circuitos impresos sin tener que retirarlos de su sitio.
15. Chip de comunicación que permite la conversión de serial a USB.

### 6.2.1.3 Entorno de programación o IDE

Arduino nos proporciona un entorno para la programación, este es un IDE liviano, con herramientas básicas para subir, depurar y comunicar con la placa Arduino. (del Valle Hernández, 2018)

Las herramientas que nos proporciona el IDE son las siguientes:

- **Auto formato**, formatea nuestro código de tal forma que sea legible. Para proyectos pequeños, es fácil que el programador, le dé formato al código de forma sencilla, pero cuando lleva muchas líneas de código, esta opción es muy útil.
- **Archivo de programa**, comprime toda la carpeta del proyecto para guardarla en un fichero ZIP.
- **Reparar codificación y Recargar**, esta opción es útil para reparar ficheros con diferentes codificaciones. A todos nos ha pasado, que nos han enviado un fichero UTF8 y lo hemos guardado con una codificación diferente y todos los caracteres raros, en fin, se vuelven raros :-D. Esto nos permite reparar los ficheros.
- **Monitor serie**, es un simple monitor serie, que nos permite comunicarnos con la placa. Nos permite tanto enviar como recibir datos del Arduino, enviados a través del objeto Serial.
- **Placa y Puerto**, nos permite seleccionar con que placa estamos trabajando (modelo) y en que puerto está conectado.
- **Programador**, en caso de disponer de un programador de PIC's, podemos seleccionar uno determinado para programar nuestro chip.
- **Quemar bootloader**, nos permite volver a cargar el bootloader (programa encargado de iniciar el chip y periféricos, así como iniciar nuestro programa).

### 6.2.1.4 Lenguaje de programación

El lenguaje de programación base es c++.



## **6.2.2 Servicios web**

Un Servicio Web es un componente al que podemos acceder mediante protocolos Web estándar, utilizando XML para el intercambio de información. (IA, 2013) Normalmente nos referimos con Servicio Web a una colección de procedimientos (métodos) a los que podemos llamar desde cualquier lugar de Internet o de nuestra intranet, siendo este mecanismo de invocación totalmente independiente de la plataforma que utilicemos y del lenguaje de programación en el que se haya implementado internamente el servicio. Cuando conectamos con un servidor web desde nuestro navegador, el servidor nos devuelve la página web solicitada, que es un documento que se mostrará en el navegador para que lo visualice el usuario, pero es difícilmente entendible por una máquina. Podemos ver esto como web para humanos. En contraposición, los Servicios Web ofrecen información con un formato estándar que puede ser entendido fácilmente por una aplicación. En este caso estaríamos ante una web para máquinas. Los servicios Web son componentes de aplicaciones distribuidas que están disponibles de forma externa. Se pueden utilizar para integrar aplicaciones escritas en diferentes lenguajes y que se ejecutan en plataformas diferentes.

### **6.2.2.1 Características del servicio web**

Las características deseables de un Servicio Web son: (IA, 2013)

- Un servicio debe poder ser accesible a través de la Web. Para ello debe utilizar protocolos de transporte estándares como HTTP, y codificar los mensajes en un lenguaje estándar que pueda conocer cualquier cliente que quiera utilizar el servicio.
- Un servicio debe contener una descripción de sí mismo. De esta forma, una aplicación podrá saber cuál es la función de un determinado Servicio Web, y cuál es su interfaz, de manera que pueda ser utilizado de forma automática por cualquier aplicación, sin la intervención del usuario.
- Debe poder ser localizado. Debemos tener algún mecanismo que nos permita encontrar un Servicio Web que realice una determinada función. De esta forma tendremos la posibilidad de que una aplicación localice el servicio que necesite de forma automática, sin que el usuario tenga que conocerlo previamente.

### **6.2.2.2 Características del servicio web**

Las características deseables de un Servicio Web son: (IA, 2013)

- Un servicio debe poder ser accesible a través de la Web. Para ello debe utilizar protocolos de transporte estándares como HTTP, y codificar los mensajes en un lenguaje estándar que pueda conocer cualquier cliente que quiera utilizar el servicio.
- Un servicio debe contener una descripción de sí mismo. De esta forma, una aplicación podrá saber cuál es la función de un determinado Servicio Web, y cuál es su interfaz, de manera que pueda ser utilizado de forma automática por cualquier aplicación, sin la intervención del usuario.
- Debe poder ser localizado. Debemos tener algún mecanismo que nos permita encontrar un Servicio Web que realice una determinada función. De esta forma tendremos la posibilidad de que una aplicación localice el servicio que necesite de forma automática, sin que el usuario tenga que conocerlo previamente.

### **6.2.2.3 Tipos de servicio web**

A nivel conceptual, un servicio es un componente software proporcionado a través de un endpoint accesible a través de la red. (IA, 2013) Los servicios productores y consumidores utilizan mensajes para intercambiar información de invocaciones de petición y respuesta en forma de documentos auto-contenidos que hacen muy pocas asunciones sobre las capacidades tecnológicas de cada uno de los receptores.

A nivel técnico, los servicios pueden implementarse de varias formas. En este sentido, podemos distinguir dos tipos de servicios Web: los denominados servicios Web "grandes" ("big" Web Services), los llamaremos servicios Web SOAP, y servicios Web RESTful.

### **6.2.2.4 Servicios web SOAP**

Los servicios Web SOAP, o servicios Web "big", utilizan mensajes XML para comunicarse que siguen el estándar SOAP (Simple Object Access Protocol), un lenguaje XML que define la arquitectura y formato de los mensajes. (IA, 2013) Dichos sistemas normalmente contienen una descripción legible por la máquina de la descripción de las operaciones ofrecidas por el servicio, escrita en WSDL (Web Services Description Language), que es un lenguaje basado en XML para definir las interfaces sintácticamente. El formato de mensaje SOAP y el lenguaje de definición

de interfaces WSDL se ha extendido bastante, y muchas herramientas de desarrollo, por ejemplo, NetBeans, pueden reducir la complejidad de desarrollar aplicaciones de servicios Web. El diseño de un servicio basado en SOAP debe establecer un contrato formal para describir la interfaz que ofrece el servicio Web. WSDL puede utilizarse para describir los detalles del contrato, que pueden incluir mensajes, operaciones, bindings, y la localización del servicio Web. También deben tenerse en cuenta los requerimientos no funcionales, como por ejemplo las transacciones, necesidad de mantener el estado (addressing), seguridad y coordinación. En este módulo vamos a hablar únicamente en los Servicios Web SOAP.

#### **6.2.2.5 Servicios web RESTful**

Los servicios Web RESTful (Representational State Transfer Web Services) son adecuados para escenarios de integración básicos ad-hoc. (IA, 2013) Dichos servicios Web se suelen integrar mejor con HTTP que los servicios basados en SOAP, ya que no requieren mensajes XML o definiciones del servicio en forma de fichero WSDL. Los servicios Web REST utilizan estándares muy conocidos como HTTP, SML, URI, MIME, y tienen una infraestructura "ligera" que permite que los servicios se construyan utilizando herramientas de forma mínima. Gracias a ello, el desarrollo de servicios RESTful es barato y tiene muy pocas "barreras" para su adopción.

#### **6.2.2.6 Arquitectura de los servicios web**

Los servicios Web presentan una arquitectura orientada a servicios que permite crear una definición abstracta de un servicio, proporcionar una implementación concreta de dicho servicio, publicar y localizar un servicio, seleccionar una instancia de un servicio, y utilizar dicho servicio con una elevada interoperabilidad. (IA, 2013) Es posible desacoplar la implementación del servicio Web y su uso por parte de un cliente. También es posible desacoplar la implementación del servicio y de cliente. Las implementaciones concretas del servicio pueden desacoplarse a nivel de lógica y transporte.

### 6.2.2.6 Volley

Volley es una librería desarrollada por Google para optimizar el envío de peticiones Http desde las aplicaciones Android hacia servidores externos. (Revelo, 2015) Este componente actúa como una interfaz de alto nivel, liberando al programador de la administración de hilos y procesos tediosos de parsing, para permitir publicar fácilmente resultados en el hilo principal. Volley es un cliente Http creado para facilitar la comunicación de red en las aplicaciones Android. Está totalmente enfocado en las peticiones, evitando la creación de código repetitivo para manejar tareas asíncronas por cada petición o incluso para parsear los datos que vienen del flujo externo.

Entre sus características más potenciadoras podemos encontrar:

- Procesamiento concurrente de peticiones.
- Priorización de las peticiones, lo que permite definir la preponderancia de cada petición.
- Cancelación de peticiones, evitando la presentación de resultados no deseados en el hilo principal.
- Gestión automática de trabajos en segundo plano, dejando de lado la implementación manual de un framework de hilos.
- Implementación de caché en disco y memoria.
- Capacidad de personalización de las peticiones.
- Provee información detallada del estado y flujo de trabajo de las peticiones en la consola de depuración.

Volley posee varios componentes que optimizan la administración de las peticiones generadas desde las aplicaciones Android. La gestión comienza en una Cola de Peticiones que recibe cada una de las peticiones generadas, donde son previamente priorizadas para su realización.

Luego son seleccionadas por un elemento llamado Cache Dispatcher, cuya función es comprobar si la respuesta de la petición actual puede ser obtenida de resultados previos guardados en caché. Si es así, entonces se pasa a parsear la respuesta almacenada y luego se presenta al hilo principal. En caso negativo, se envía la petición a la Cola de Conexiones Pendientes, donde reposan todas aquellas peticiones que están por ejecutarse.

Luego entra en juego un componente llamado Network Dispatcher, el cual se encarga de seleccionar las peticiones pendientes de la cola, para realizar las respectivas transacciones Http hacia el servidor. Si es necesario, la respuesta de estas peticiones se guarda en caché, luego se parsean y finalmente se publican en el hilo principal.

### **6.2.3 PHP**

PHP<sup>2</sup> (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. PHP está enfocado principalmente a la programación de scripts del lado del servidor, por lo que se puede hacer cualquier cosa que pueda hacer otro programa CGI, como recopilar datos de formularios, generar páginas con contenidos dinámicos, o enviar y recibir cookies

### **6.2.4 MYSQL**

MySQL<sup>3</sup> es la base de datos de código abierto número uno del mundo, es la base de datos número uno para Web y es una excelente base de datos embebida. Más de 3.000 ISVs y OEMs, incluyendo 8 de los 10 mayores, y 17 de los 20 principales proveedores de software de todo el mundo confían en MySQL como base de datos de sus productos. En este documento se revisan las diez razones técnicas por qué MySQL es capaz de satisfacer las necesidades de muchos proveedores líderes, así como miles de nuevas empresas, para hacer funcionar sus aplicaciones, que van desde telecomunicaciones y la seguridad, a Contabilidad de pequeñas empresas y Educación. El documento analiza muchas de las nuevas características clave de MySQL, e incluye numerosos ejemplos de clientes ISV y OEM.

---

<sup>2</sup> Puedes saber más de PHP vía <http://php.net/>

<sup>3</sup> para saber más de MYSQL visita <https://www.mysql.com/why-mysql/white-papers/las-10-razones-principales-para-usar-mysql-como-base-de-datos-integrada/>

### 6.2.5 MpAndroidChart

MPAndroidChart, es una librería que nos permite mostrar de forma fácil y personalizable grandes conjuntos de datos en forma de gráficas. (marjimlao, 2015) De esta forma el usuario disfruta de una mejor experiencia de uso.

Entre los tipos de gráfica que se pueden usar están:

LineChart (gráfico de línea), BarChart (gráfico de barras), PieChart (gráfico circular), ScatterChart (Gráfico de dispersión), CandleStickChart (gráfico de candelero), BubbleChart (Gráfico de burbuja), RadarChart (Gráfico de radar)

Para utilizar los datos en una gráfica antes se debe incluir en un objeto del tipo *DataSet*. Dependiendo del tipo de gráfica a utilizar necesitaremos un tipo de *DataSet*. Por ejemplo, para una gráfica del tipo *BarChart* usaremos un objeto del tipo *BarDataSet*. De la misma forma, para una gráfica del tipo *PieChart* usaremos un objeto del tipo *PieDataSet*.

Con estos datos se puede llenar el tipo de gráfica que se desee, la gráfica que se genera es interactiva y responde a los movimientos para acercar y arrastrar.

Esta biblioteca le permite personalizar completamente la posible interacción táctil (y gesticular) con la vista de gráfico y reaccionar a la interacción a través de los métodos de devolución de llamada. (Jay, 2017)

Habilitar / deshabilitar la interacción

- `setTouchEnabled(boolean enabled)`: Permite habilitar / deshabilitar todas las interacciones táctiles posibles con el gráfico.
- `setDragEnabled(boolean enabled)`: Habilita / deshabilita el arrastre (panorámica) para el gráfico.
- `setScaleEnabled(boolean enabled)`: Habilita / deshabilita la escala para el gráfico en ambos ejes.
- `setScaleXEnabled(boolean enabled)`: Habilita / deshabilita la escala en el eje x.
- `setScaleYEnabled(boolean enabled)`: Habilita / deshabilita la escala en el eje y.
- `setPinchZoom(boolean enabled)`: Si se establece en verdadero, el zoom de pellizco está habilitado. Si está desactivado, los ejes xey se pueden ampliar por separado.
- `setDoubleTapToZoomEnabled(boolean enabled)`: Establezca esto en falso para no permitir el acercamiento del gráfico haciendo doble clic en él.

## Apertura / desaceleración del gráfico

- `setDragDecelerationEnabled(boolean enabled)`: Si se establece en verdadero, el gráfico continúa desplazándose después de retocar. Predeterminado: verdadero
- `setDragDecelerationFrictionCoef(float coef)`: Coeficiente de fricción de desaceleración en [0; 1] intervalo, los valores más altos indican que la velocidad disminuirá lentamente; por ejemplo, si se establece en 0, se detendrá inmediatamente. 1 es un valor no válido, y se convertirá a 0.9999 automáticamente.

## Metodos para resaltar las entradas en el gráfico:

### Activando / Desactivando el resaltado

- `setHighlightPerDragEnabled(boolean enabled)`: Establezca esto en verdadero Chart para permitir el resaltado por arrastrar sobre la superficie del gráfico cuando está completamente alejada. Predeterminado: verdadero
- `setHighlightPerTapEnabled(boolean enabled)`: Configure esto como falso Chart para evitar que los valores se resalten con un toque del gesto. Los valores todavía se pueden resaltar mediante arrastre o mediante programación. Predeterminado: verdadero
- `setMaxHighlightDistance(float distanceDp)`: Establece la distancia máxima de resaltado en dp. Los toques en la tabla más alejados de una entrada que esa distancia no activarán un punto culminante. Predeterminado: 500dp

## Metodos de configuración y estilo aplicable para todos los Charttipos.

### Refrescante

- `invalidate()`: Llamar a este método en el gráfico lo **actualizará** (redibujará). Esto es necesario para que los cambios realizados en el gráfico tengan efecto.
- `notifyDataSetChanged()`: Permite que el gráfico sepa que los datos subyacentes han cambiado y realiza todos los cálculos necesarios (compensaciones, leyenda, máximos, mínimos, ...). Esto es especialmente necesario cuando se agregan datos dinámicamente .

### Explotación florestal

- `setLogEnabled(boolean enabled)`: Al establecer esto en verdadero se activará la salida del logcat del gráfico. Permitir esto es malo para el rendimiento, mantenga desactivado si no es necesario.

## **Diseño general de gráficos**

Aquí hay algunos métodos generales de estilo que puede usar directamente en el cuadro:

- `setBackgroundColor(int color)`: Establece el color de fondo que cubrirá toda la vista de gráfico. Además, se puede establecer un color de fondo. Xml en el archivo de diseño.
- `setDescription(String desc)`: Establece un texto de descripción que aparece en la esquina inferior derecha del gráfico.
- `setDescriptionColor(int color)`: Establece el color del texto de descripción.
- `setDescriptionPosition(float x, float y)`: Establece una posición personalizada para el texto de descripción en píxeles en la pantalla.
- `setDescriptionTypeface(Typeface t)`: Establece el Typeface utilizado para dibujar el texto descriptivo.
- `setDescriptionTextSize(float size)`: Establece el tamaño del texto de descripción en píxeles, min 6f, max 16f.
- `setNoDataText(String text)`: Establece el texto que debería aparecer si el gráfico está vacío.
- `setDrawGridBackground(boolean enabled)`: Si está habilitado, se dibujará el rectángulo de fondo detrás del área de dibujo del gráfico.
- `setGridBackgroundColor(int color)`: Establece el color con el que debe dibujarse el fondo de la cuadrícula.
- `setDrawBorders(boolean enabled)`: Activa / desactiva el dibujo de los bordes del gráfico (líneas que rodean el gráfico).
- `setBorderColor(int color)`: Establece el color de las líneas del borde del gráfico.
- `setBorderWidth(float width)`: Establece el ancho de las líneas del borde del gráfico en dp.
- `setMaxVisibleValueCount(int count)`: Establece el número máximo de etiquetas de valor dibujadas visibles en el gráfico. Esto solo tiene efecto cuando `setDrawValues()` está habilitado.

## **7. Procedimiento de las Actividades Realizadas**

### **7.1 Análisis del diseño del prototipo**

Con base en nuestro proyecto de residencia se realizó un recorrido en el invernadero del instituto tecnológico de Tuxtla Gutiérrez con la finalidad de conocer la forma en la que se iniciaría la experimentación de los gases en un ambiente controlado y en el cual se utilizaría el prototipo de detección de gases para la lectura del mismo. Los gases utilizados en la experimentación son una mezcla de diferentes gases los cuales son benceno, xileno y tolueno.



Durante el desarrollo de la residencia, se realizaron actividades tales como la instalación del prototipo de detección de gases en un pequeño contenedor, la codificación del prototipo y la inspección de la obtención de datos de lectura del gas.

Hasta el momento las experimentaciones se realizan en los contenedores de acrílico hechas por compañeros de la carrera de química, en las cuales se encuentran plantas de la familia de las bromelias las cuales son utilizadas por los compañeros del área de química para la evaluación de la capacidad de la planta de Bioacumular Hidrocarburos Cíclicos (BTX), siendo utilizado como un bioindicador<sup>45</sup>; la información obtenida es utilizada para conocer la concentración de los gases dentro de las cajas.

Como parte de nuestro proyecto, nos corresponde realizar las experimentaciones. Debido a que, por información proporcionada por los ingenieros de química, en el horario de 5 a 9 pm la planta mantiene mucho más abiertos sus poros que la hacen consumir mucho más alimento, que en este caso es la mezcla BTX<sup>6</sup>, las pruebas se hacen en ese rango de horas.

Es importante mencionar que el proyecto está siendo realizado por las ingenierías en Sistemas Computacionales y maestría del área de Química.

De manera paralela a estas experimentaciones, estamos trabajando en las aplicaciones para los dispositivos móviles; se creó un servicio web el cual recibirá los datos del prototipo y los guardará para después enviar esa información a las aplicaciones las cuales graficarán los datos. Este servicio web se encuentra alojado en un servidor gratuito para el uso de pruebas en las aplicaciones, después de terminar las pruebas se alojará en un servidor de pago de mayor confianza.

Las aplicaciones serán creadas en Android Studio y Swift dependiendo del dispositivo al cual está dirigido, las dos aplicaciones se encuentran conectadas al servicio web y ya reciben datos de este.

## **7.2 Calibración de los sensores**

Los sensores utilizados para el prototipo son de alta sensibilidad, el rango de detención es de 10 a 1000 ppm en el caso de benceno y en los gases xileno y tolueno son de 10 a 900 ppm. Estos sensores contienen una resistencia la cual al entrar en contacto con el gas genera un dato llamado resistencia de sensibilidad (Rs).

---

<sup>5</sup> Organismo vivo que se utiliza para determinar y evaluar el índice de contaminación de un lugar

<sup>6</sup> Mezcla de los Compuestos Benceno, Tolueno y Xileno.

La calibración de estos sensores se lleva a cabo al medir la resistencia de sensibilidad inyectando en un contenedor hermético una cantidad controlada de los gases censados, el dato obtenido se asocia al dato obtenido en la resistencia de sensibilidad del sensor.

### **7.3 Programación de módulo**

El módulo utilizado para el prototipo es el esp8266 el cual soporta una conexión IPv4 y los protocolos TCP, el módulo cuenta con tres modos de actividad los cuales son Active mode, Sleep mode y Deep sleep, en esta ocasión se configuro el modo de Active mode con el cual el módulo trabajara a pleno rendimiento en todo momento.

En la configuración del módulo de escribe el ssid de la red y la contraseña de la misma, de esta manera el sensor al estar encendido se conectará enseguida, la siguiente configuración fue establecer los datos que se enviaran desde el módulo y el destino al el cual lo enviara.

Aquí podemos observar una parte del código donde establecemos una conexión TCP con nuestro servidor web.

```
EnvioDeDatos ("AT+CIPSTART=\"TCP\", \"\"javierbtx.000webhostapp.com\" \"\",80\"\"r\n\",1000, DEBUG);  
// Establece una conexión TCP o registra un puerto UDP e inicia la conexión
```

### **7.4 Desarrolló de los servicios web**

En la programación del servicio web se creó una base de datos con una sola tabla en la cual se guardan los datos provenientes del prototipo, los cuales son (Fecha, Sensor de Gas, Sensor de Temperatura y Humedad) esta tabla es llenada por medio de un archivo PHP por medio de un método GET.

El servicio web se encuentra alojado en un servidor gratuito para la fase de pruebas una vez terminado el proyecto se migrará a un servidor mejor.



## 8. Resultados obtenidos

### 8.1 Fase de Análisis

#### 8.1.1 Requisitos Funcionales de la aplicación móvil

**RF1.** El usuario podrá visualizar los datos almacenados en la base de datos, en forma de tabla.

**RF2.** El usuario visualizar una gráfica de los datos censados

**RF3.** El usuario podrá generar un archivo PDF que contendrá la información de los datos censados en forma de tabla y la gráfica de estos.

**RF4.** El usuario podrá dar zoom a la gráfica.

**RF5.** El usuario podrá desplazar horizontal y verticalmente las vistas de la aplicación.

#### 8.1.2 Descripción de actor(es)

<b>Nombre</b>	Usuario
<b>Casos de uso</b>	Datos obtenidos de los sensores Ver gráficas Generar PDF
<b>Tipo</b>	Primario
<b>Descripción</b>	El usuario es el actor que podrá tener libre acceso a todas las funciones implementadas en la aplicación Android (AppBTX).

*Cuadro 1: Descripción de actor Usuario.*

### 8.1.3 Diagrama de casos de uso

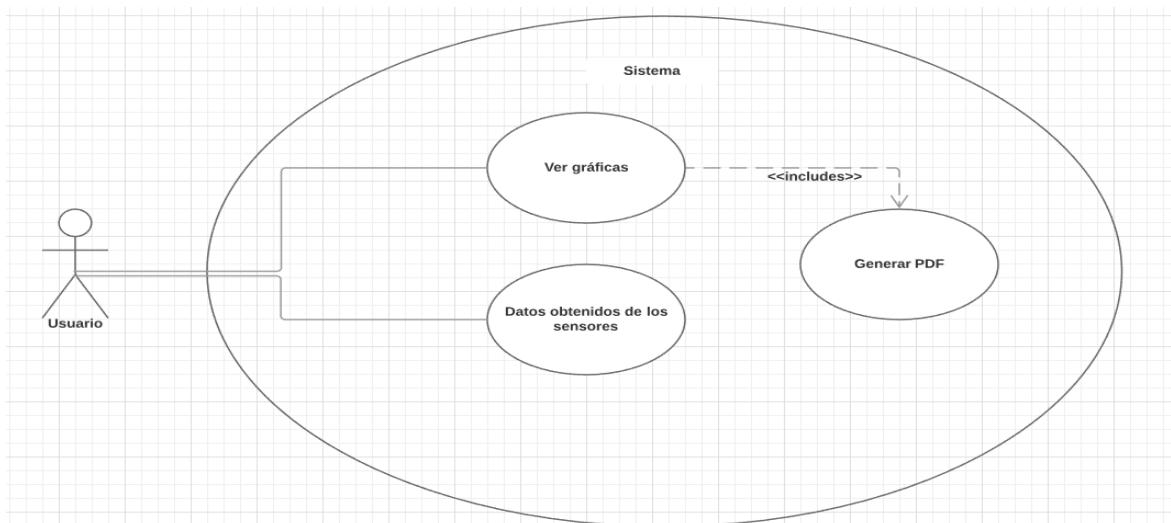


Figura 6: Diagrama de casos de uso: Usuario. basada en (Sommerville (2010))

### 8.1.4 Plantillas de casos de uso

<b>Nombre del caso de uso</b>	Datos obtenidos de los sensores	
<b>ID del caso de uso</b>	CDU-1	
<b>Fuente</b>	Requisito Funcional: RF1, RF5	
<b>Actor(es)</b>	Usuario	
<b>Propósito</b>	Visualizar datos censados que se encuentran dentro del servidor web.	
<b>Precondición</b>	Debe de haber datos censados dentro del servidor web.	
<b>Ocasionador</b>	Se inicia cuando el usuario quiere visualizar la información sobre los datos censados que hay en el servidor.	
<b>Flujo de eventos</b>		
<b>Actor(es)</b>	<b>Sistema</b>	
1		Muestra la vista principal de la aplicación. (figura 16)
2	Selecciona la opción Datos obtenidos de los sensores.	
3		Muestra la pantalla con la información de los datos censados que se encuentran dentro del servidor, en forma de tabla. (figura 18)
4	Desplaza horizontal y verticalmente la pantalla	
5		se visualiza en la pantalla la información que estaba fuera del rango de

	resolución del dispositivo móvil (si es el caso).
<b>Curso de salida</b>	Salir
<b>Curso alternativo</b>	Si no se puede establecer conexión con el servidor, se mostrará el mensaje “no se pudo establecer conexión”.

*Cuadro 2: Descripción del caso de uso: Datos obtenidos de los sensores.*

<b>Nombre del caso de uso</b>	Ver gráficas
<b>ID del caso de uso</b>	CDU-2
<b>Fuente</b>	Requisito Funcional: RF2, RF4
<b>Actor(es)</b>	Usuario
<b>Propósito</b>	Visualizar una gráfica de los datos obtenidos por los sensores para ver su comportamiento de una manera óptima.
<b>Precondición</b>	Debe de haber datos censados dentro del servidor web.
<b>Ocasionador</b>	Se inicia cuando el usuario quiere ver en una gráfica el comportamiento de los datos censados.
<b>Flujo de eventos</b>	
<b>Actor(es)</b>	<b>Sistema</b>
1	Muestra la vista principal de la aplicación. (figura 16)
2	Selecciona la opción Ver gráficas.
3	Muestra la pantalla con la información de los datos censados en forma de gráfica. (figura 20)
4	Desplaza dos puntos en sentido contrario uno de otro en la pantalla.
5	Se realiza un zoom en la gráfica para la mejor visualización de la parte requerida.
<b>Curso de salida</b>	Salir

*Cuadro 3: Descripción del caso de uso: Ver graficas.*

<b>Nombre del caso de uso</b>	Generar PDF
<b>ID del caso de uso</b>	CDU-3
<b>Fuente</b>	Requisito Funcional: RF3
<b>Actor</b>	Usuario
<b>Propósito</b>	Generar un archivo PDF que contenga una tabla y una gráfica de los datos censados.
<b>Precondición</b>	El usuario debe de haber entrado a la opción Ver gráficas
<b>Ocasionador</b>	El usuario desea archivar la información de los datos censados.
<b>Flujo de eventos</b>	
<b>Actor</b>	<b>Sistema</b>
1	Muestra la pantalla con la información de los datos censados en forma de gráfica. (figura 20)
2	Selecciona el menú ítem
3	Muestra en la pantalla el menú ítem. (figura 22)
4	Selecciona la opción Generar PDF
5	Se genera y muestra el archivo PDF.
<b>Curso</b>	Salir
<b>Curso alternativo</b>	Si el sistema móvil no cuenta con un lector de archivos PDF el documento no será abierto

Cuadro 4: Descripción del caso de uso: Generar PDF.

### 8.1.5 Requisitos no funcionales

- 1.- El sistema debe de poseer interfaces gráficas bien formadas
- 2.- La interfaz del sistema será de fácil comprensión.
- 3.- El sistema debe tener un tiempo de respuesta adecuado.

#### 8.1.5.1 Como se obtuvo los requisitos no funcionales

- 1.- Para la interfaz gráfica de la aplicación se siguieron las reglas de Material design, gracias a ello se logró un diseño agradable al usuario.
- 2.- Para lograr este punto se le dio una descripción entendible a cada una de las pantallas.

3.- Para lograr un tiempo de respuesta adecuado en cuanto a la obtención de datos del servidor se optó por usar la librería Volley ya que como se mencionó en el marco teórico Volley optimiza las peticiones al servidor creando así una respuesta rápida, en cuanto a la graficación de los datos se optó por usar la librería Mpandroidchart ya que tiene un tiempo de respuesta rápida.

## 8.2 Fase de diseño

### 8.2.1 Esquema conceptual del sistema

Este diagrama representa la estructura general de cómo está compuesto el sistema a desarrollado.

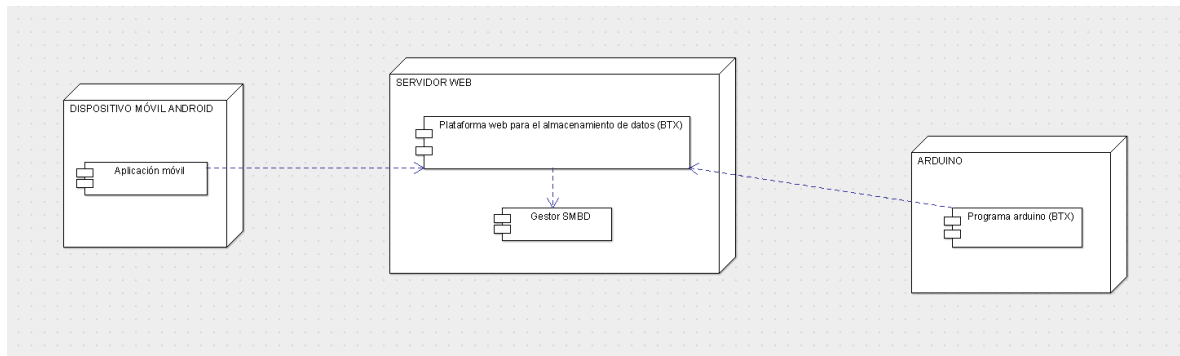


Figura 7 Diagrama de Despliegue. Creación propia.



## 8.2.2 Diagrama de clases de la aplicación

Este diagrama representa las clases y métodos a usar en la aplicación móvil, se puede observar la relación que habrá en cada una de ellas

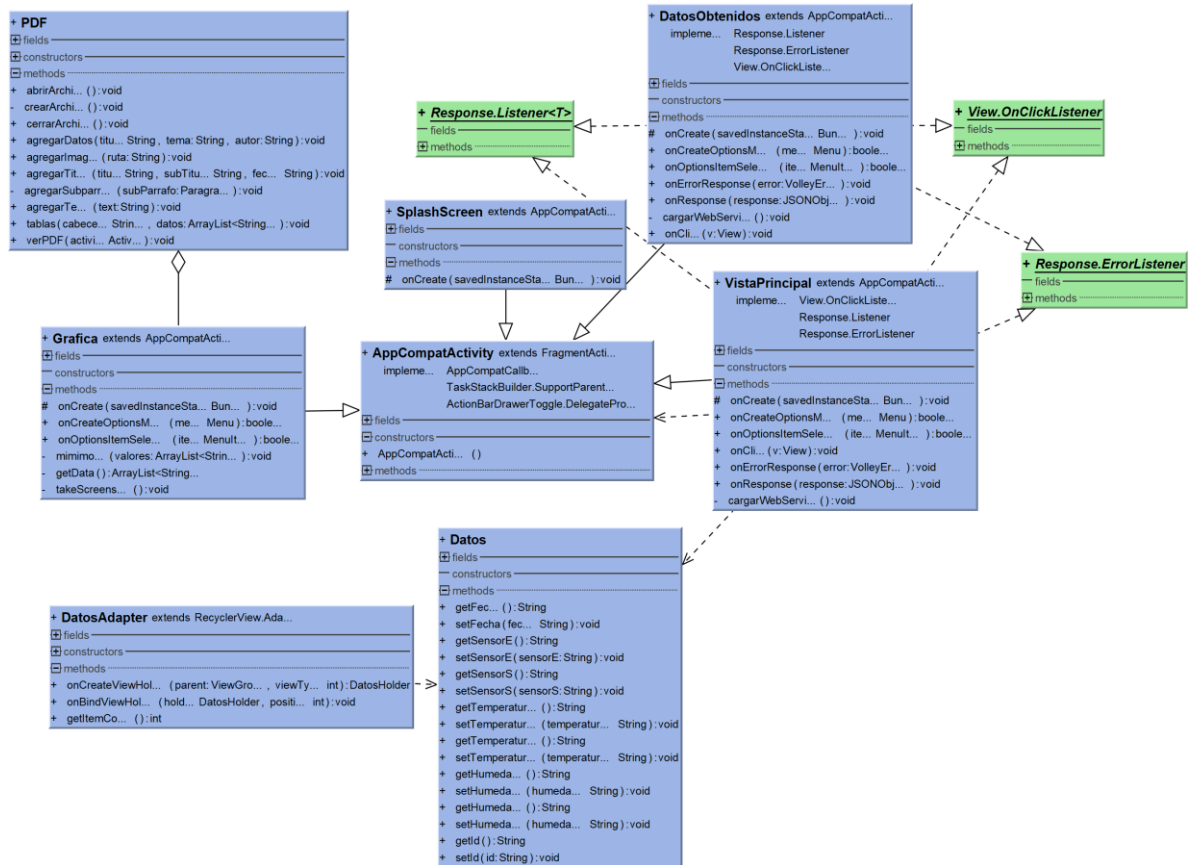
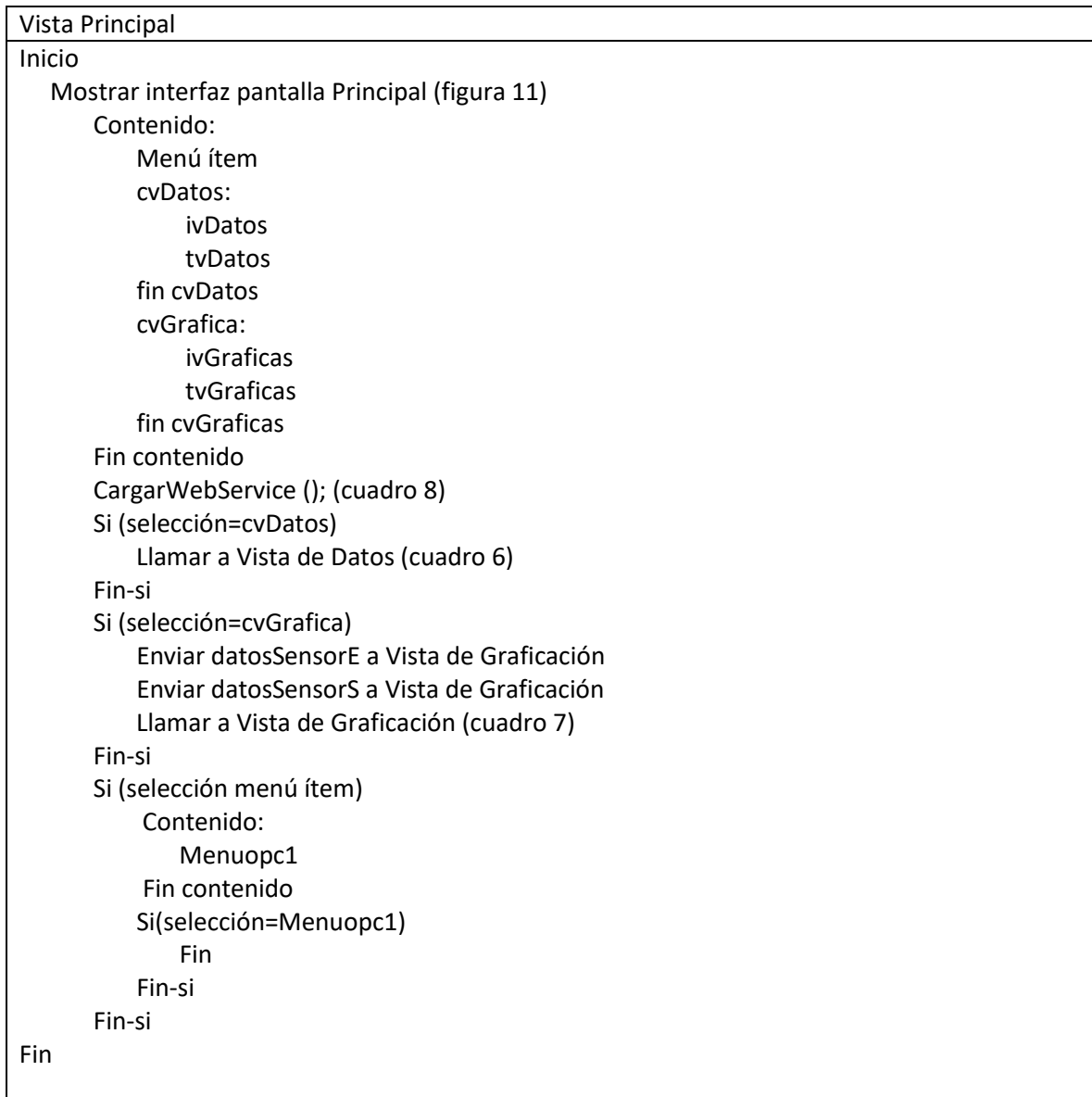


Figura 8: Diagrama de clases de la aplicación

## 8.2.3 Diseño detallado de la aplicación móvil



Cuadro 5: Algoritmo: vista principal.

Inicio

Mostrar interfaz pantalla de datos (figura 13)

Contenido:

Menú ítem

TxtFecha

TxtSensorE

TxtSensorS

TxtTempE

TxtTempS

TxtHumdE

TxtHumdS

RecyclerView

Fin contenido

CargarWebService (); (cuadro 8)

Si (selección menú ítem)

Contenido:

Menuopc1

Fin contenido

Si(selección=Menuopc1)

Llamar Vista Principal (cuadro 5)

Fin-si

Fin-si

Fin

*Cuadro 6: Algoritmo: Vista de Datos*

Vista de Graficación
<p>Inicio</p> <p>Mostrar interfaz pantalla Principal (figura 11)</p> <p>Contenido:</p> <p>    lcSenseo</p> <p>Fin contenido</p> <p>Recibir datosSensorE</p> <p>Recibir datosSensorS</p> <p>Para (i=1 hasta, datosSensor,1)</p> <p>    Agregar a lcSenseo (datosSensorE,datosSensorS)</p> <p>Fin Para</p> <p>Si (selección menú ítem)</p> <p>    Contenido:</p> <p>        Menuopc1</p> <p>        Menuopc2</p> <p>    Fin contenido</p> <p>    Si(selección=Menuopc1)</p> <p>        Llamar PDF (); (cuadro 9)</p> <p>    Fin-si</p> <p>    Si(selección=Menuopc2)</p> <p>        Llamar Vista Principal (cuadro 5)</p> <p>    Fin-si</p> <p>Fin-si</p> <p>Fin</p>

*Cuadro 7: Algoritmo: Vista de Graficación*

CargarWebService ();
<p>Inicio</p> <p>    Si (respuesta=true)</p> <p>        Realiza una Consulta</p> <p>        Tabla: datosSenseo</p> <p>        Retorna estado de la consulta</p> <p>    Fin-si</p> <p>    Si-No</p> <p>        Imprimir “no se pudo establecer conexión”</p> <p>    Fin-no</p> <p>Fin</p>

*Cuadro 8: Algoritmo: Cargar Web Service ();*

PDF ();

Inicio

    Crear archivo:

        Si (creación = cierta)

            Crear tabla:

                Si (creación =cierta)

                    Agregar al archivo

                Fin-si

            Si-No

                Imprimir "error al crear tabla"

            Fin-no

        Fin-Crear tabla

    Crear Imagen:

        Si (creación =cierta)

            Agregar al archivo

        Fin-si

        Si-No

            Imprimir "error al crear Imagen"

        Fin-no

    Fin-Crear tabla

Fin-si

Si-No

    Imprimir "error al crear archivo"

    Fin-no

Fin-Crear archivo

Fin

*Cuadro 9: Algoritmo: PDF ();*

## 8.2.4 Programación de la placa Arduino

### 8.2.4.1 Diagrama de flujo del programa de Arduino

En este diagrama se representa la lógica del programa de Arduino para las acciones que debe realizar conforme a la detección y envío de datos.

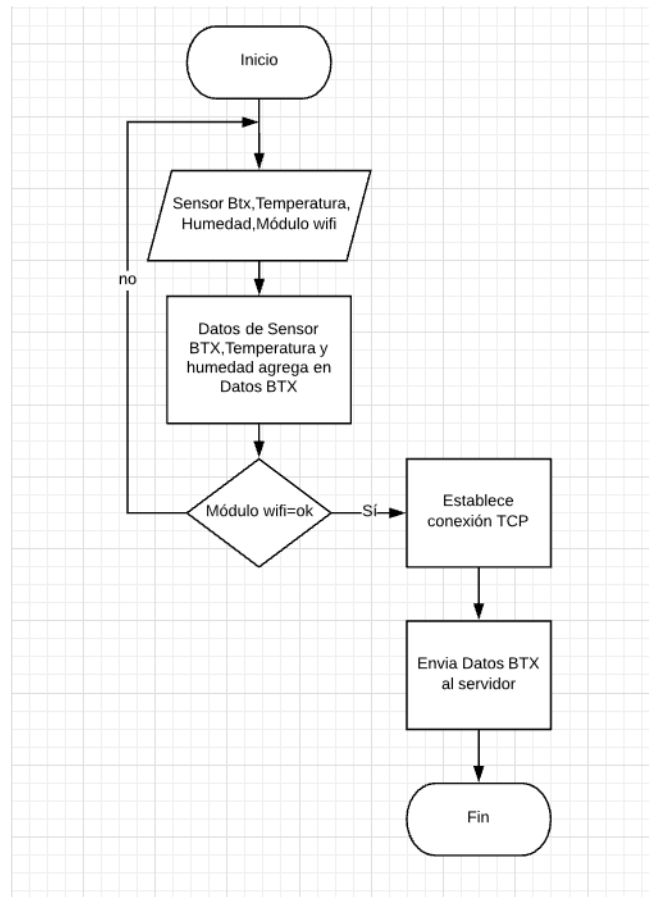


Figura 9: Diagrama de flujo del programa de Arduino.  
Creación propia.

1. Inicio
2. Recibe en las variables Sensor BTX, Temperatura, Humedad, Módulo wifi.
3. Los datos de las variables Sensor BTX, Temperatura y Humedad se agregan en la variable Datos BTX como una tabla.
4. Si la variable Módulo wifi = ok, Entonces: Establece conexión TCP con el servidor
5. Envía los datos de la variable BTX al servidor
6. De lo contrario: Regresa a la opción 2
7. Fin.

## 8.2.5 Programación de los servicios web

Para tener un mejor manejo de los datos censados se requería que estos tuvieran una forma de almacenamiento, por lo cual se generó una base de datos en el servidor para que el Arduino pueda enviar los datos a esa base y la aplicación pueda recibirlos.

### 8.2.5.1 Desarrollo de la base de datos

La base de datos está constituida por una sola tabla para la recepción de los datos provenientes de las lecturas de los sensores en el prototipo.

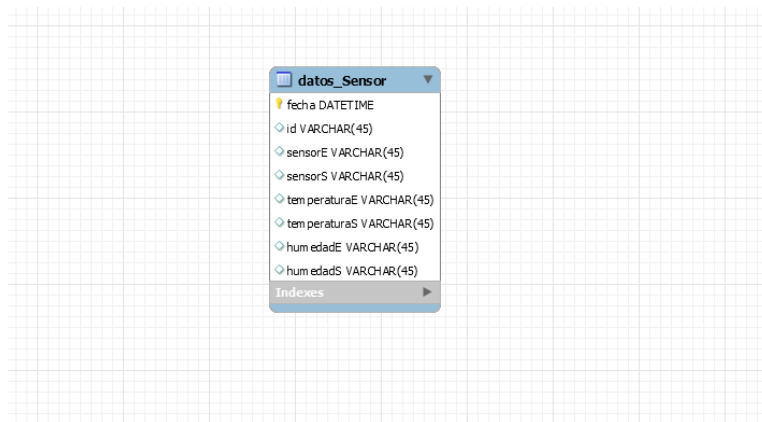


Figura 10: Base de datos

Nombre del campo	Tipo de campo	Descripción
fecha	DATETIME	Fecha en la cual se tomó el dato
id	VARCHAR (45)	Es el identificador de donde se toma el dato
sensorE	VARCHAR (45)	Es el sensor de entrada
sensorS	VARCHAR (45)	Es el sensor de salida
temperaturaE	VARCHAR (45)	Es el sensor de la temperatura de entrada
temperatueaS	VARCHAR (45)	Es el sensor de la temperatura de salida
humedadE	VARCHAR (45)	Es el sensor de humedad de entrada
humedadS	VARCHAR (45)	Es el sensor de humedad de salida

Cuadro 10: Descripción de campos de la base de datos

### 8.2.5.2 Diagrama de flujo primer servicio

En este diagrama de flujo es el servicio en el cual se obtienen los datos del arduino y los almacena en la base de datos del servidor web.

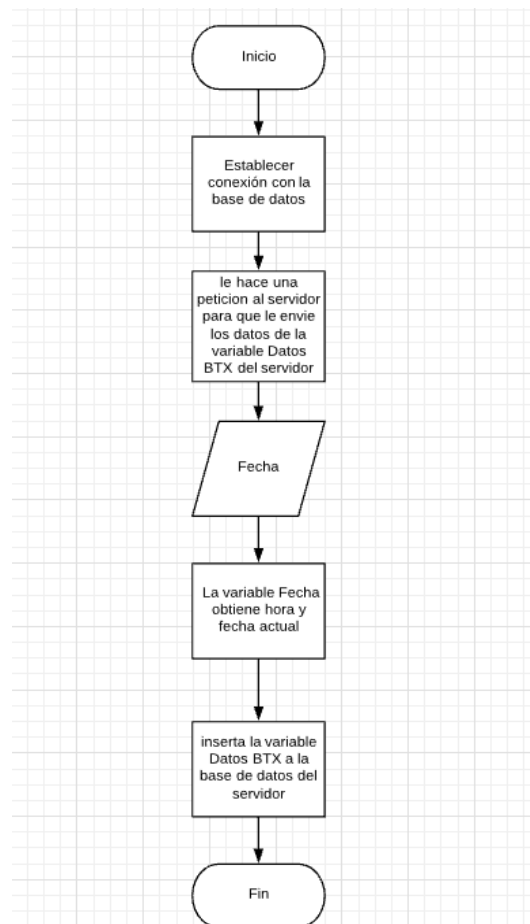


Figura 11: Diagrama de flujo para insertar datos al servidor. Creación propia.



1. Inicio.
2. Establecer conexión con la base de datos.
3. Realizar una petición al servidor para que le envíe los datos de la variable “Datos BTX” del servidor.
4. Crea la variable fecha.
5. La variable fecha obtiene la hora y fecha actual del servidor.
6. Inserta la variable datos BTX a la base de datos del servidor.
7. Fin.

### 8.2.5.3 Diagrama de flujo segundo servicio

En este diagrama de flujo es el servicio en el cual se puede consultar los datos que están almacenados en el servidor web.

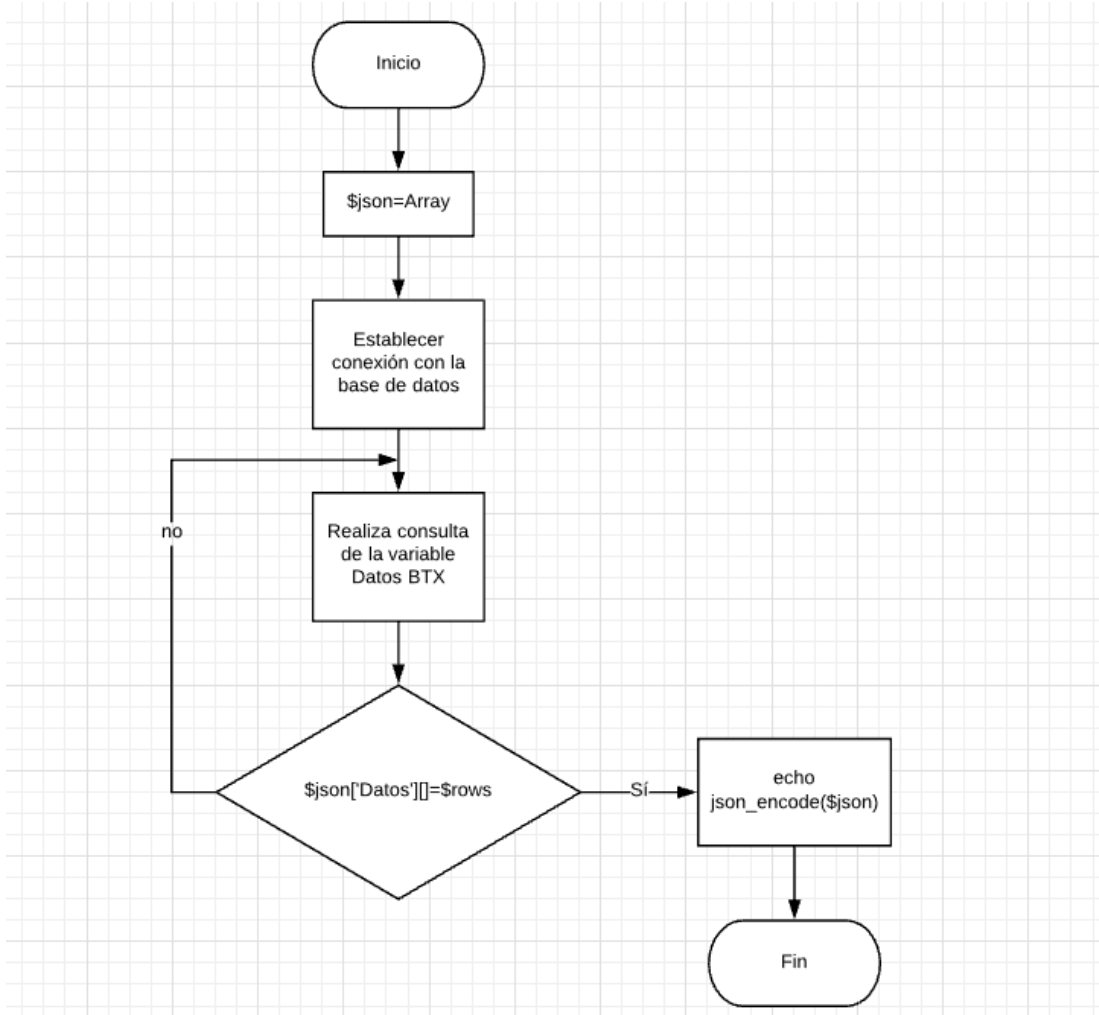


Figura 12: Diagrama de flujo para la consulta de datos en el servidor. Creación propia.

1. Inicio.
2. La variable \$json se iguala a la variable array ().
3. Se establece conexión con la base de datos.
4. Realiza consulta de la variable "Datos BTX".
5. Si la variable \$json["Datos"][]=\$rows, Entonces: se almacena la variable "Datos BTX" en el arreglo
6. Se realiza "echo json\_encode(\$json)" para poder ver los datos en el arreglo
7. De lo contrario: se regresa al paso número 3.

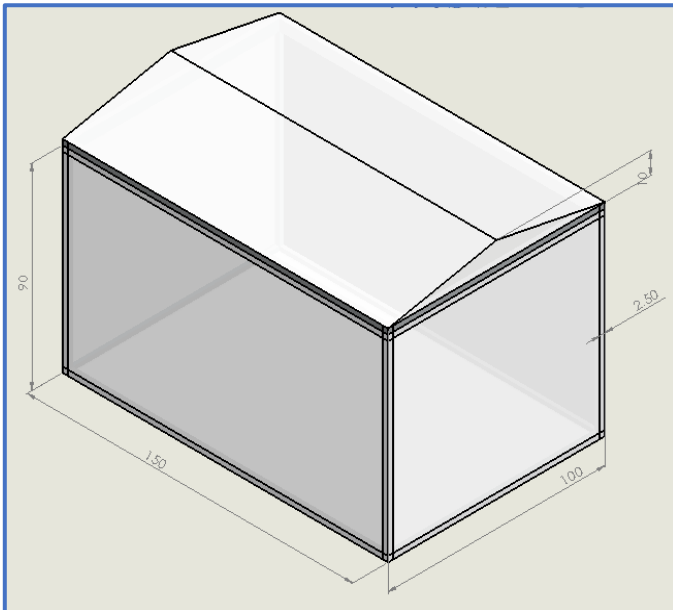
## 9. Resultados finales

La recopilación de la información de la lectura de los sensores se hace de lunes a viernes de 5 pm a 9 pm, se conectan los sensores a sus respectivos arduinos la cual está conectada a una computadora, una vez conectado el sensor se espera un tiempo para la estabilización de los sensores, se considera estabilizado un sensor cuando inicia a arrojar la misma lectura consecutivamente pues la resistencia del sensor a quemado los residuos de gas este proceso lleva de 4 a 5 minutos en cada sensor utilizado. La información es recabada en una macro creada en un libro de Excel donde se encuentra la información de los sensores de gas y temperatura.

Una vez que los sensores se encuentran estables se procede a cerrar de manera hermética los contenedores de acrílico las cuales contienen dentro un recipiente con el compuesto BTX (Benceno, Tolueno y Xileno) el cual una vez cerrado se procede a nebulizar por 10 minutos, una vez pasado el tiempo se recogen los datos de la concentración dentro de la caja, para revisar la presencia del gas.

Una vez recopilado el dato de lectura de presencia de gas se abren las cajas para liberar el gas restante.

### 9.1 Diseño de Contenedor de Acrílico



Las medidas del contenedor son:

Altura: 90 cm.

Largo: 150 cm.

Ancho: 100 cm.

Este contenedor tiene un volumen de 1.42 m<sup>3</sup>.

Figura 13: Caja de Acrílico

## 9.2 Diseño del prototipo

El prototipo fue creado en base a las necesidades del proyecto los componentes del prototipo son: (Figura 14).

1. Placa Arduino uno.
2. Sensor de gas Tolueno.
3. Sensor de gas Xileno y Benceno.
4. Módulo Wifi.
5. Sensor de Temperatura y Humedad.
- 6.

### 9.2.1 Esquema del prototipo

El prototipo debe de enviar los datos de lectura a un servidor web donde se alojarán y podrán ser visualizados desde la app de Android.

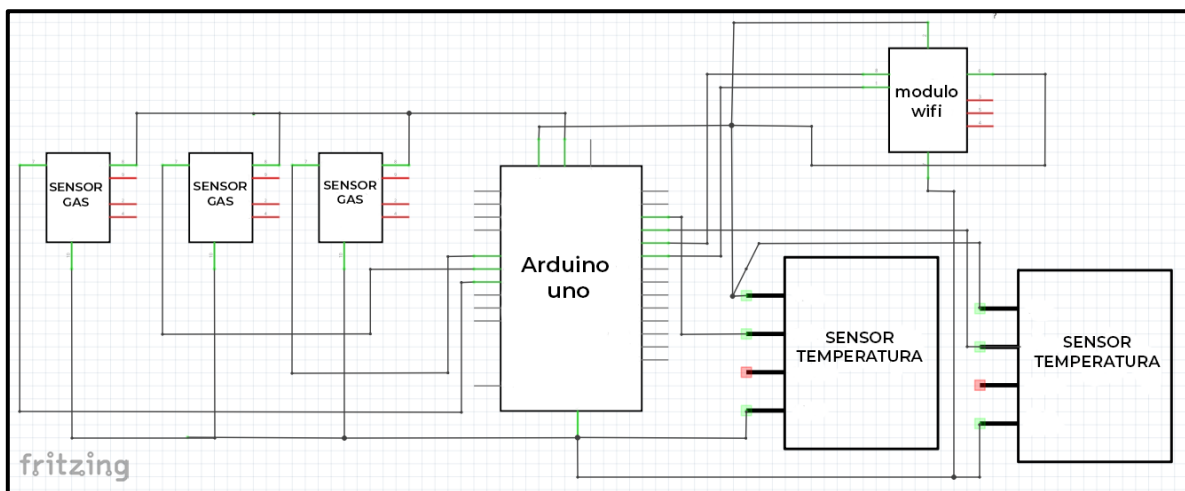


Figura 14: Esquema del prototipo.

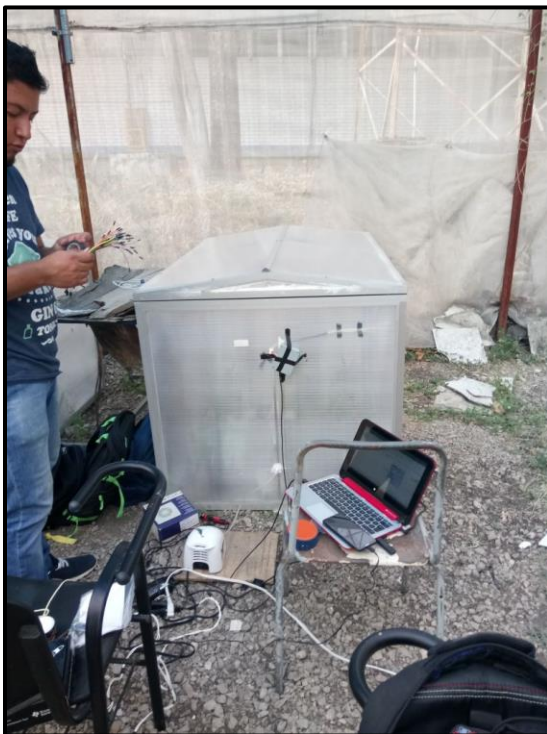
## 9.3 Vistas de la aplicación y del prototipo

### 9.3.1 Vista de colocación del prototipo en contenedor acrílico

Podemos observar en la Imagen 1 el nebulizador con el cual se inyectaba en forma de gas el compuesto BTX al contenedor de acrílico.



*Imagen 1: Imagen del nebulizador*



*Imagen 2: Conexión de los sensores*

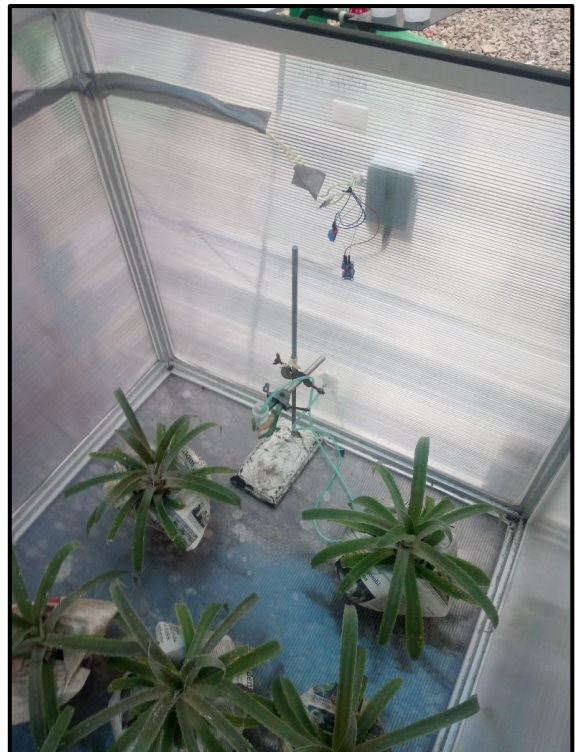
En la Imagen 2 se está revisando las conexiones de los sensores y se estaba censando.



*Imagen 3: Plantas Bromelias*

En estas imágenes podemos observar en interior de los contenedores de acrílico y las plantas bromelias

Aquí podemos observar los sensores de temperatura y gas conectados al prototipo pegado al contenedor de acrílico.



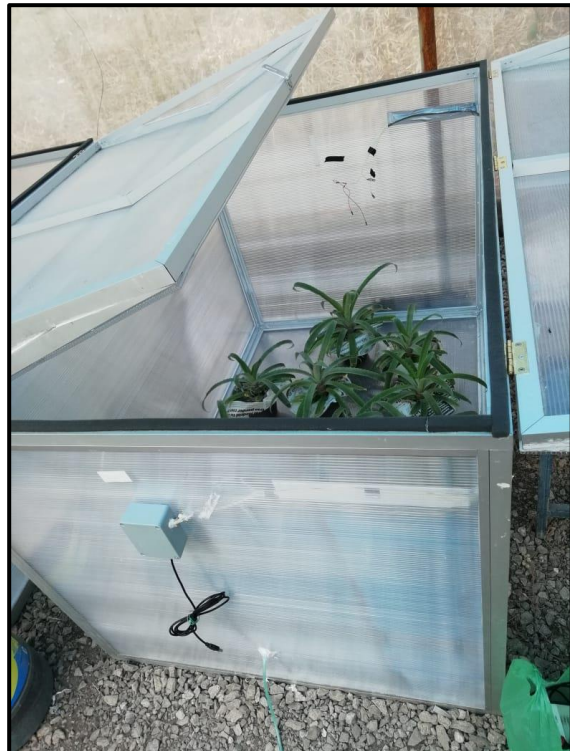
*Imagen 4: Sensores de temperatura y gas*



*Imagen 5: Prototipo Arduino*

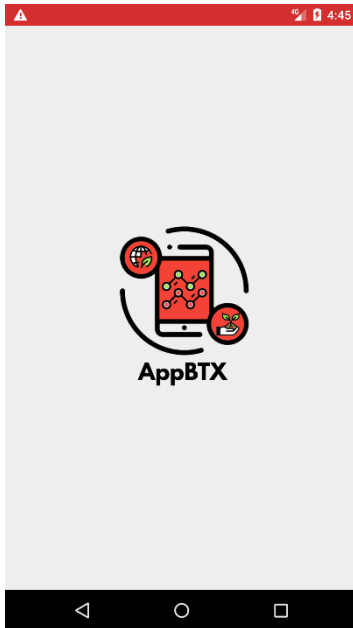
En esta imagen podemos observar el prototipo de Arduino dentro de su caja que lo protege del agua y demás.

En esta imagen podemos observar el contenedor de acrílico y las conexiones de los sensores.



*Imagen 6 Conexión de los sensores*

### 9.3.2 Vistas de la aplicación Android



En esta imagen podemos observar la que el usuario observa cuando abre la aplicación y carga por primera vez.

Figura 15: Pantalla de carga

En esta pantalla podemos observar las acciones que el usuario puede realizar que son ver las gráficas o datos obtenidos de los sensores.



Figura 16: Inicio



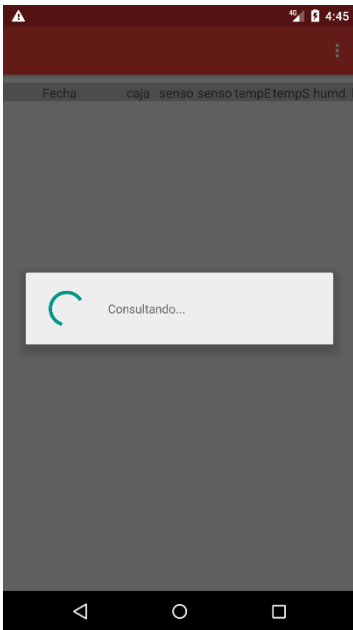


Figura 17: Mensaje de Conexión

En esta imagen podemos observar el mensaje de conexión de la aplicación a nuestro servidor web, para la visualización de los datos.

En esta pantalla podemos observar los datos recopilados por el sensor en forma de una tabla de datos, en la parte superior podemos observar la opción de graficar. Se puede observar de manera vertical y horizontal.

 A screenshot of an Android application showing a horizontal table of sensor data. The table has 8 columns: Fecha, caja, sensorE, sensorS, tempE, tempS, humdE, and humdS. The data is displayed in a light gray background with black text. The table is oriented horizontally on the screen.
 

Fecha	caja	sensorE	sensorS	tempE	tempS	humdE	humdS
2018-05-12 14:22:15	C1	022	064	28	28	45	45
2018-05-12 14:23:47	C1	039	097	28	28	44	45
2018-05-12 14:24:34	C1	086	279	28	28	44	45
2018-05-12 14:25:03	C1	083	267	28	28	44	45
2018-05-12 14:25:52	C1	081	258	28	28	44	45
2018-05-12 14:26:20	C1	074	250	28	28	44	45
2018-05-12 14:27:16	C1	073	249	28	28	44	45
2018-05-12 14:28:02	C1	072	248	28	28	44	45
2018-05-12 14:28:33	C1	070	246	28	28	44	45
2018-05-12 14:28:40	C1	070	247	28	28	44	45
2018-05-12 14:35:40	C1	069	246	28	28	44	45
2018-05-12 14:36:17	C1	067	244	28	28	44	45
2018-05-12 14:39:53	C1	065	243	28	28	44	45
2018-05-12 14:40:54	C1	064	242	28	28	44	45

Figura 19: Tabla Horizontal

 A screenshot of an Android application showing a vertical table of sensor data. The table has 8 columns: Fecha, caja, senso, senso tempE tempS humd. The data is displayed in a light gray background with black text. The table is oriented vertically on the screen.
 

Fecha	caja	senso	senso	tempE	tempS	humd	humd
2018-05-12 14:22:15	C1	022	064	28	28	45	45
2018-05-12 14:23:47	C1	039	097	28	28	44	45
2018-05-12 14:24:34	C1	086	279	28	28	44	45
2018-05-12 14:25:03	C1	083	267	28	28	44	45
2018-05-12 14:25:52	C1	081	258	28	28	44	45
2018-05-12 14:26:20	C1	074	250	28	28	44	45
2018-05-12 14:27:16	C1	073	249	28	28	44	45
2018-05-12 14:28:02	C1	072	248	28	28	44	45
2018-05-12 14:28:33	C1	070	246	28	28	44	45
2018-05-12 14:28:40	C1	070	247	28	28	44	45
2018-05-12 14:35:40	C1	069	246	28	28	44	45
2018-05-12 14:36:17	C1	067	244	28	28	44	45
2018-05-12 14:39:53	C1	065	243	28	28	44	45
2018-05-12 14:40:54	C1	064	242	28	28	44	45
2018-05-12 14:43:27	C1	064	242	28	28	44	45
2018-05-12 14:43:53	C1	062	239	28	28	44	45
2018-05-12 14:44:24	C1	062	239	28	28	44	45
2018-05-12 14:44:55	C1	061	240	28	28	44	45
2018-05-12 14:45:16	C1	061	240	28	28	44	45
2018-05-12 14:46:24	C1	060	240	28	28	44	45
2018-05-12 14:46:34	C1	060	241	28	28	44	45
2018-05-12 14:47:52	C1	059	239	28	28	44	45
2018-05-12 14:48:06	C1	058	240	28	28	44	45
2018-05-12 14:48:26	C1	057	240	28	28	44	45
2018-05-12 14:48:41	C1	058	240	28	28	44	45
2018-05-21 00:00:00	C1	058	240	28	28	44	45
2018-05-21 00:00:00	C1	058	240	28	28	44	45

Figura 18: Tabla Vertical

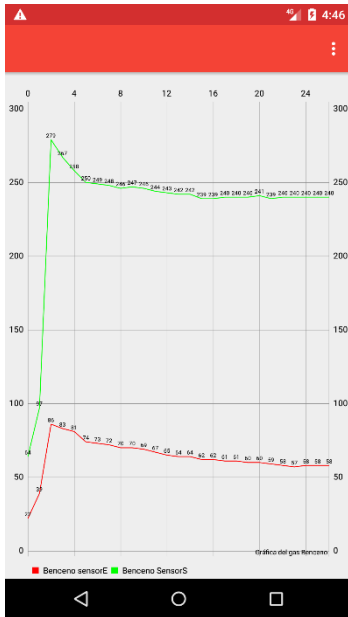


Figura 20: Grafica

En esta pantalla podemos observar la opción de graficar los datos, en este ejemplo se muestra la comparación de lecturas entre dos sensores de benceno, cada uno de los sensores cuenta con una línea de puntos de diferentes colores para su fácil identificación.

En esta imagen podemos observar los mismos datos de manera horizontal

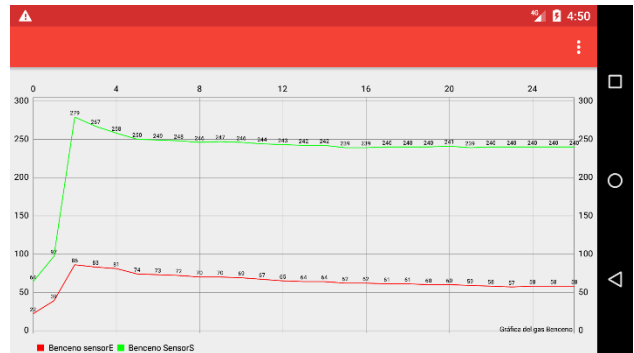
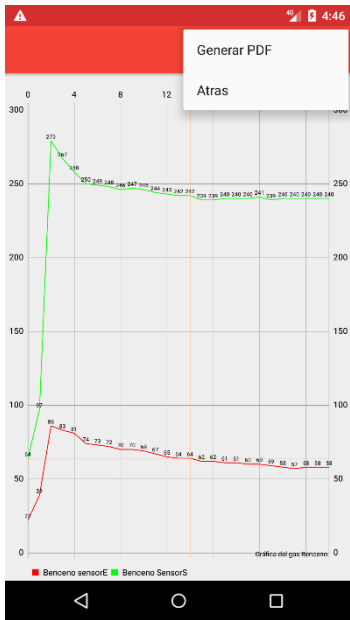


Figura 21: Grafica horizontal



En esta imagen podemos ver el menú que tiene a la hora de abrir la gráfica y poder generar un PDF con los datos de los sensores.

Figura 22: Generar PDF

En esta imagen podemos observar las capturas del archivo PDF que se genera al darle clic en la opción.

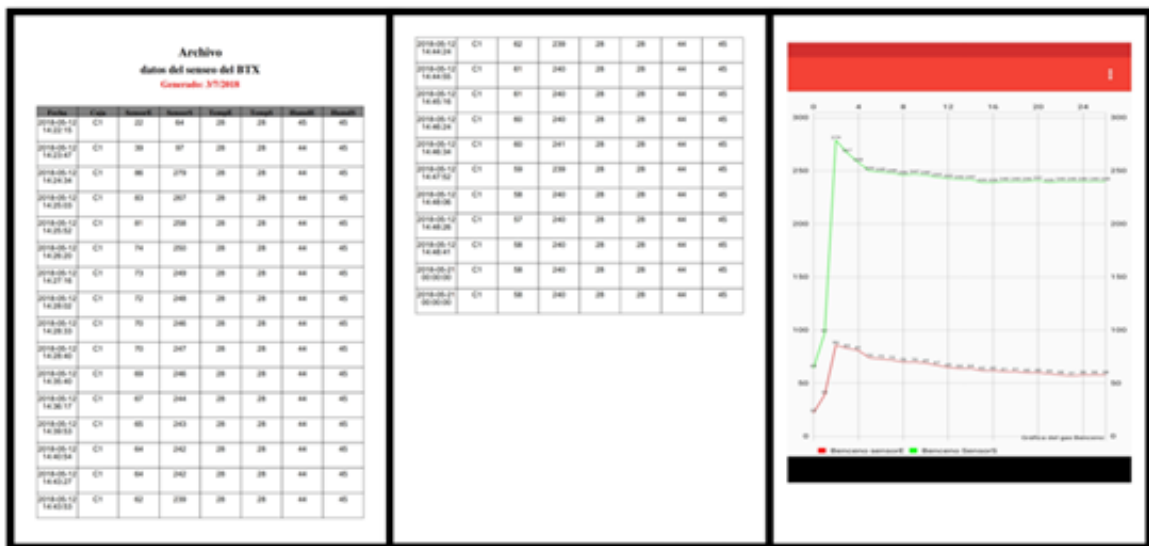


Figura 23: Capturas PDF

## 10. Conclusión

En este proyecto se logró aplicar nuestras habilidades como ingenieros en sistemas en un problema de otro ámbito un poco alejado al nuestro como lo es la bioquímica

Todo esto no hubiese sido posible sin la colaboración de las ingenierías tanto de Sistemas Computacionales como la Ingeniería en Bioquímica, la parte fundamental fue poder realizar todo el prototipo para su correcta implementación en un ambiente controlado y mediante el análisis que día a día se fueron tomando, poder obtener una base de datos de los hidrocarburos que fueron utilizados (Benceno, Tolueno y Xileno).

Para obtener los datos de los sensores utilizados en este proyecto se tuvo que obtener los datos mediante el análisis del comportamiento de los gases a través del uso de un nebulizador para la extracción del gas y poder esparcirlo en los contenedores de acrílico y así poder obtener la calibración de los sensores para su correcto uso.

El diseño de la aplicación se platicó con el asesor externo y se sugirió que los datos que el sensor detectaran fueran en forma de tabla descendente para la mejor visualización de los datos que se analizaron y poder observarlos en una gráfica para obtener una mejor vista del comportamiento de los hidrocarburos.

Al finalizar el proyecto obtuvimos gran conocimiento de los sensores de gas que se utilizaron y que fueron de gran ayuda con lo que se quería lograr ya que de estos se obtenían los datos que se compararon con lo que se le agregaba al compuesto siempre que analizábamos el comportamiento de los gases.

## **11. Competencias desarrolladas**

### **11.1 Competencias genéricas**

1. Coordinar y participar en proyectos interdisciplinarios.
2. Diseñar e implementar interfaces hombre-máquina y máquina-máquina para la automatización de sistemas.
3. Identificar y comprender las tecnologías de hardware para proponer, desarrollar y mantener aplicaciones eficientes.
4. Seleccionar y aplicar herramientas matemáticas para el modelado, diseño y desarrollo de tecnología computacional.

### **11.2 Competencias desarrolladas en la elaboración de la aplicación móvil**

- Manejo de librerías
- Conocimiento de conexión con el servicio web
- Manipulación de múltiples datos
- Conocimiento de las reglas de material design para el diseño de la aplicación
- Manejo de archivos no editables (PDF)

### **11.3 Competencias desarrolladas en la elaboración del prototipo basado en Arduino**

- Manejo de librerías para los sensores utilizados en el proyecto
- Manipulación de sensores de alta precisión
- Creación de servicios wifi utilizando comandos AT (Arduino)
- Calibración de sensores

## 12. Fuentes de Información

- Cuevas, D. (s.f.). M. D. C PRODUCCION DE BTX EN MÉXICO: USOS, TOXICOLOGÍA Y ANÁLISIS.
- del Valle Hernández, L. (05 de julio de 2018). *programafacil.com*. Obtenido de <https://programafacil.com/podcast/28-entorno-de-desarrollo-de-arduino/>
- IA, C. d. (06 de 2013). *jtech*. Obtenido de <http://www.jtech.ua.es/j2ee/publico/servc-web-2012-13/wholesite.pdf>
- Jay, P. (15 de mayo de 2017). *GitHub*. Obtenido de <https://github.com/PhilJay/MPAndroidChart/wiki>
- marjimlao. (08 de octubre de 2015). *Tutoriales Android*. Obtenido de <https://tutorialesandroidweb.wordpress.com/2015/09/08/graficas-en-android-con-mpandroidchart/>
- Mexicana. (1993). NORMA Oficial Mexicana NOM-047-SSA1-1993, Que establece los límites biológicos máximos permisibles de disolventes orgánicos en el personal ocupacionalmente expuesto. *Mexicana*.
- Mexicana. (1994). NOM-127-SSA1-1994, salud ambiental. Agua para uso y consumo humano. Límites permisibles de calidad y tratamientos a que debe someterse el agua para su potabilización, México, Secretaría de Salud. *Mexicana*.
- Mexicana. (2003). Límites máximos permisibles de hidrocarburos en suelos y las especificaciones para su caracterización y remediación. *Mexicana*.
- Orozco, B., Pérez, S., & González, D. (2003). *Contaminación ambiental. Una Visión Desde la Química*. Barcelona: Thomsom.
- Profeyoni. (15 de Febrero de 2015). *Diseño y manufactura*. Obtenido de <https://www.hermosaprogramacion.com/2015/02/android-volley-peticiones-http/>
- Revelo, J. (Febrero de 2015). *Hermosa programación*. Obtenido de <http://www.hermosaprogramacion.com/2015/02/android-volley-peticiones-http/>
- Sommerville, I. (2010). *Ingeniería de Software*. Madrid (España): Pearson Addison Wesley.
- Toledo, M. (s.f.). PRODUCCIÓN DE BTX EN MEXICO: USOS, TOXICOLOGÍA Y ANALISIS BTX PRODUCED IN MEXICO: USES, TOXICOLOGY AND ANALYSES.
- Vollhardt. (2007). *Química orgánica: estructura y función*. Omega.