



# **UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

---

**FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN**

## **UN MODELO DE REFERENCIA PARA EL DISEÑO DE SOLUCIONES BASADO EN EL PARADIGMA DE INTERNET DE LAS COSAS**

**TESIS**

**QUE PARA OBTENER EL TÍTULO DE:  
LICENCIADO EN INFORMÁTICA**

**PRESENTA:**

**RUIZ REYES AYRTON DANIEL**

**ASESOR:**

**MTIA. GENARO ACEVEDO GARCIA**



**Ciudad Universitaria, CD. MX., 2016**



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



Dedico este trabajo a Ana Bertha Reyes por todo tu amor, esfuerzo y trabajo dedicados a lo largo de estos 24 años, porque en los malos momentos siempre vas a estar ahí para escucharme e intentar aconsejarme y por ser el apoyo incondicional de mis malas decisiones. Gracias, Madre, por ser la guía bondadosa que nunca se rindió por más obscuro que fuera el pronóstico.

A Ricardo Reyes por ser el mejor maestro que alguien podrá tener en una vida, por enseñarme que el error y los sacrificios no son malo si se aprenden de ellos, que todas las acciones tienen sus consecuencias y demostrarme todo lo que puede lograrse con esfuerzo y dedicación. Gracias hermano, por estar ahí siempre.

A mi mascota, Lestat, que fue el acompañante de las largas noches en las que este documento fue desarrollado, escrito y revisado. No existe un gato mejor.

Gracias a mis amigos, porque esta tesis tiene un poco de ustedes en él. Gracias por escucharme y debatir, por los ánimos, los momentos buenos y malos, los festejos, los días y las noches. Porque lo importante son los lazos que formamos y los momentos compartidos, y yo soy la serie de experiencias vividas y por vivir que me permiten compartir.

Al Mtro. Genaro Acevedo por la confianza depositada en mí y la oportunidad de llevar a la realidad una idea diferente. Por la motivación, los retos, las conversaciones y el tiempo dedicado. Muchas gracias.

A la Universidad Nacional Autónoma de México, porque pertenecer a la máxima casa de estudios es un honor que muy poco poseemos, espero que esta aportación sea de utilidad para generaciones venideras, y pueda devolver un poco de todo lo que me has brindado.

A quienes roban sonrisas, para construir mejores historias.

Gracias totales.

Ayrton Daniel Ruiz Reyes

## **Agradecimientos**

Quiero agradecer al Dr. Armando Maldonado Talamantes por la asesoría durante el desarrollo de este documento, por la confianza en el potencial del conocimiento aquí presentado y la oportunidad de transmitirlo en el curso que el imparte dentro del Instituto Tecnológico Autónomo de México.

Al Dr. Guillermo Aguilar Sahagún por los consejos y comentarios durante la revisión de este documento, y por todo el apoyo brindado durante estos meses.

A ambos, por la dedicación al guiarme a convertir este intento de novela en un documento de investigación, por las bromas y sus invaluable opiniones sobre el tema y asuntos relacionados. Muchas gracias.

A Gabriela Escalante, la dueña del invernadero en donde se montó una de las pruebas de concepto desarrolladas, por las facilidades brindadas y la atención durante las conversaciones que complementaron la idea inicial.

Finalmente, a María del Carmen Pérez Medina y a Lorena Guadalupe Martínez Bárcenas, pues las investigaciones derivadas de la prueba de concepto del Invernadero sobre indicadores e inteligencia de negocios complementaron la documentación del mismo.

# Índice

<b>Introducción</b> .....	4
<b>Capítulo 1 INTERNET DE LAS COSAS</b> .....	6
<b>Conceptos básicos</b> .....	6
<b>¿Qué es un paradigma?</b> .....	6
<b>Paradigma orientado a objetos</b> .....	8
<b>Bases de datos</b> .....	9
<b>Minería de datos</b> .....	9
<b>Sistemas de Inteligencia de Negocios</b> .....	10
<b>Internet</b> .....	10
<b>Metodología OpenUp</b> .....	11
<b>Internet de las Cosas</b> .....	16
<b>La estructura funcional del IoT</b> .....	20
<b>Modelo Machine-to-Machine (M2M)</b> .....	21
<b>Actuales plataformas que apoyan proyectos IoT</b> .....	22
<b>Plataformas en la nube</b> .....	22
<b>Windows 10 IoT</b> .....	24
<b>Capítulo 2 DISPOSITIVOS EN EL HARDWARE LIBRE</b> .....	26
<b>¿Qué es el hardware?</b> .....	26
<b>Iniciativas open source o libres</b> .....	27
<b>Diferencia entre microcontroladores y microprocesadores</b> .....	28
<b>Arduino</b> .....	30
<b>Historia</b> .....	30
<b>Principales Productos y Características</b> .....	31
<b>Raspberry pi</b> .....	34
<b>Historia</b> .....	35
<b>Principales Productos y Características</b> .....	35
<b>Ejemplos de Sensores</b> .....	39
<b>Sensor DHT11</b> .....	39
<b>Sensor Humedad de Superficie</b> .....	39
<b>Capítulo 3 SOFTWARE EMPRESARIAL</b> .....	41
<b>¿Qué es el software?</b> .....	41

Breve historia de software empresarial.....	43
Clasificación del software empresarial .....	45
<b>Capítulo 4 MODELO DE REFERENCIA .....</b>	<b>47</b>
<b>Motivos y origen del modelo .....</b>	<b>47</b>
<b>Un modelo de referencia basado en capas.....</b>	<b>50</b>
<b>Seguridad .....</b>	<b>54</b>
<b>Beneficios.....</b>	<b>56</b>
<b>Iniciativas Extranjeras.....</b>	<b>56</b>
<b>Iniciativas Nacionales .....</b>	<b>57</b>
<b>Capítulo 5 PRUEBAS DE CONCEPTO .....</b>	<b>59</b>
<b>PC1: Mejora de mediciones ambientales en un invernadero .....</b>	<b>60</b>
<b>Capa de negocio y proceso .....</b>	<b>60</b>
<b>Capa de Objetos.....</b>	<b>60</b>
<b>Capa de Sensores .....</b>	<b>61</b>
<b>Capa de Pre-Procesamiento .....</b>	<b>63</b>
<b>Capa de Comunicaciones.....</b>	<b>64</b>
<b>Capa de Almacenamiento.....</b>	<b>64</b>
<b>Capa de Procesamiento .....</b>	<b>65</b>
<b>Soluciones Alternativas .....</b>	<b>67</b>
<b>Resultados.....</b>	<b>68</b>
<b>PC2: Herramienta de apoyo para sistemas de inventarios sobre líquidos.....</b>	<b>69</b>
<b>Capa de negocio y proceso .....</b>	<b>69</b>
<b>Capa de Objetos.....</b>	<b>70</b>
<b>Capa de Sensores .....</b>	<b>70</b>
<b>Capa de Pre-Procesamiento .....</b>	<b>70</b>
<b>Capa de Comunicaciones.....</b>	<b>72</b>
<b>Capa de Almacenamiento y procesamiento .....</b>	<b>73</b>
<b>Capa de Presentación .....</b>	<b>74</b>
<b>Soluciones Alternativas .....</b>	<b>74</b>
<b>Resultados.....</b>	<b>74</b>
<b>Conclusiones .....</b>	<b>76</b>
<b>Anexos.....</b>	<b>78</b>
<b>PC1 – Documento de Visión. ....</b>	<b>78</b>
<b>PC1 – Plan de Proyecto .....</b>	<b>83</b>

<b>PC1 - Especificación General de Requerimientos para el Sistema</b> .....	87
<b>PC1 – Diagrama de Casos de Uso</b> .....	92
<b>PC1 – Especificación de Casos de Uso</b> .....	93
<b>Caso de Uso: Obtener Temperatura y Humedad Ambiental</b> .....	93
<b>Caso de Uso: Obtener Humedad Superficial</b> .....	95
<b>Caso de Uso: Obtener pH Superficial</b> .....	97
<b>Caso de Uso: Enviar lecturas</b> .....	99
<b>Caso de Uso: Consulta Lecturas</b> .....	101
<b>PC1 – Diagrama de Entidad Relación</b> .....	104
<b>PC1 – Diagrama de Componentes</b> .....	105
<b>PC1 – Diagrama de Clases</b> .....	106
<b>PC1 – Extracto de Código</b> .....	107
<b>Archivo Invernadero_2.0_ino</b> .....	107
<b>Archivo GreenHouseControl.h</b> .....	108
<b>Archivo GreenHouseControl.cpp</b> .....	109
<b>PC1 – Memoria Fotográfica</b> .....	111
<b>PC2 – Documento de Visión</b> .....	115
<b>PC2 - Especificación General de Requerimientos para el Sistema</b> .....	119
<b>PC2 – Diagrama de Distribución del Almacén Principal</b> .....	124
<b>PC 2 – Diagrama de Casos de Uso</b> .....	125
<b>PC2 – Especificación de casos de uso</b> .....	126
<b>Caso de Uso: Especificar datos de extracción</b> .....	126
<b>Caso de Uso: Extraer producto</b> .....	128
<b>Caso de Uso: Enviar Datos</b> .....	130
<b>Caso de Uso: Registrar extracción</b> .....	132
<b>PC2 – Diagrama Entidad Relación</b> .....	134
<b>PC2 – Diagrama de Componentes</b> .....	135
<b>Bibliografía</b> .....	136

## Introducción

El objetivo del presente trabajo es el de proponer un método sobre el cual poder desarrollar soluciones tecnológicas utilizando como base un concepto emergente llamado "Internet de las cosas". Así mismo, se pretende ofrecer una alternativa al desarrollo de sistemas que se puede encontrar en el ámbito empresarial, el cual ha ido auxiliándose de componentes externos a una computadora, tales como los lectores de códigos de barras que simplifican tareas manuales rutinarias, aplicación de CCTV, entre otros.

El contenido del documento está dividido en 5 capítulos, los primeros tres abarcan conocimientos que engloba el método aquí propuesto, los últimos dos consisten en la descripción y prueba del método, a continuación, se lista una breve descripción de cada capítulo, a modo de una introducción a cada uno.

En el capítulo uno se presentan algunas definiciones del área de Tecnologías de la información a modo de ayuda para que se familiarice con conceptos que se utilizarán en los capítulos siguientes. Se introduce al tema de "Internet de las Cosas", concepto que representa la base del método, en donde se describen sus características y los componentes que lo conforman.

El capítulo dos se centra en el componente físico, proporciona un panorama general sobre dos de las grandes plataformas dentro de las tecnologías dirigidas a Internet de las Cosas: Arduino y Raspberry Pi. De ambas se revisa el concepto que las fundamenta, su historia y las características generales de su producto principal a la fecha de este trabajo, por lo que es posible que este varíe según el tiempo en el que este documento sea consultado. Como cierre de este capítulo se muestran algunos sensores que sirven como ejemplo de cómo es que los fabricantes han adaptado sus productos para ser utilizados junto a estas plataformas.

El capítulo tres se centra en el software empresarial, diseñado para cubrir las necesidades diarias de una organización. Revisamos su historia, la cual va muy ligada al avance de la computación moderna, acompañada de una clasificación que permita identificarlos. Esta clasificación es una herramienta

guía para las soluciones que se pueden desarrollar con el modelo aquí propuesto, pues permite tener un punto de partida según la necesidad que deseemos atender.

En el capítulo cuatro se proporciona la descripción del modelo de referencia: un diseño en capas con responsabilidades definidas, que permite la comunicación con la capa superior, inferior o ambas que tiene como propósito el ofrecer un orden durante la construcción de una solución. Así mismo se describe el papel de cada capa en el modelo y los beneficios de la aplicación del modelo.

El capítulo cinco trata la aplicación del método en dos pruebas de concepto que permitieron llevar esta idea plasmada en papel, a la práctica en dos entornos reales con necesidades diferentes pero resueltos con un mismo enfoque.

Como complemento de este trabajo, se incluye una sección de anexos que recaba la documentación generada por las dos pruebas de concepto, la cual incluye documentos técnicos sobre la justificación de la solución generada, descripción del funcionamiento, diagramas de apoyo para describir los flujos y el código fuente utilizado en cada una.

Finalmente, cabe resaltar que como complemento a este trabajo se encuentra publicada una ponencia con el nombre “Propuesta de un modelo de referencia basado en Internet de las Cosas para diseñar soluciones utilizando tecnologías de la información y comunicaciones” presentada durante el XX Congreso Internacional de Contaduría, Administración e Informática. Para más información se puede consultar la memoria del congreso, publicada en su página oficial<sup>1</sup>.

---

<sup>1</sup> Página oficial del XX Congreso Internacional de Contaduría, Administración e Informática, donde la memoria del evento fue registrada con el ISSN 2395-8960 [http://congreso.investiga.fca.unam.mx/es/congreso\\_xx.php](http://congreso.investiga.fca.unam.mx/es/congreso_xx.php).

# Capítulo 1

## INTERNET DE LAS COSAS

### Conceptos básicos

Antes de entrar en materia es necesario revisar algunos conceptos con el fin de esclarecer definiciones y realizar algunas precisiones en los mismos, ya que serán utilizados más adelante en este trabajo. Naturalmente se requiere de un conocimiento en algunas áreas de la computación con el fin de comprender lo expuesto aquí. Sin embargo, se pretende alentar a personas que no sean del área de Tecnologías de la Información (TI) a que exploren nuevas alternativas en su búsqueda por solucionar situaciones que puedan ocurrir en su vida profesional.

### ¿Qué es un paradigma?

En TI una parte considerable de los componentes que se utilizan día a día están basados en un paradigma: Los programas, las telecomunicaciones, las configuraciones y en general los elementos que intervienen en un sistema de información. ¿Pero a que nos referimos con esta palabra? Se entiende por paradigma el conjunto de patrones, modelos y prácticas que definen una corriente de pensamiento. En el caso de tecnologías de la información se trata de los patrones y prácticas con los cuales resolver los problemas utilizando a la tecnología como herramienta principal

En psicología el concepto es ilustrado a través de una fábula sobre cinco monos, la cual está basada en los experimentos con chimpancés de Wolfgang Köhler y algunas anotaciones del escrito de G.R. Stephenson *“Cultural acquisition of a specific learned response among rhesus monkeys”*. Dicha fábula apareció por primera vez en el libro *“Competing for the future”* de Gary Hamel & C. K. Prahalad y la podrás leer a continuación:

*“Un grupo de científicos colocó cinco monos en una jaula y en el centro de la misma una escalera con muchas bananas. Cuando uno de los monos subía por la escalera para tomar una de las bananas, los científicos lanzaban un chorro de agua fría sobre él hasta hacerle desistir en su intento.*

*Luego de algún tiempo de repetir esta operación, cuando un mono iba a subir la escalera, los otros a la fuerza se lo impedían.*

*Pasó el tiempo y ya ningún mono subía la escalera, a pesar de la tentación que significaban las bananas. Fue entonces cuando los científicos sustituyeron uno de los monos. La primera cosa que hizo el nuevo simio fue subir por la escalera, pero rápidamente los otros monos se lo impidieron por la fuerza. Después de algunas palizas, el nuevo integrante del grupo ya no subió más la escalera.*

*Un segundo mono fue sustituido y ocurrió lo mismo. El primero de los sustitutos, incluso participó con entusiasmo de la paliza que le dieron al novato para impedirle que subiera por la escalera.*

*Un tercero fue sustituido y se repitió el hecho. Finalmente, el último de los monos veteranos fue sustituido, de manera que quedaron en la jaula cinco monos que nunca habían recibido un chorro de agua fría para que desistieran de su intento de alcanzar las bananas que estaban al final de la escalera, pero que, de todas maneras, continuaban golpeando a cualquiera que intentara llegar a las bananas.*

*Estoy seguro que si fuese posible preguntarles a los monos por qué atacaban a cualquiera que intentaba subir la escalera, la respuesta sería: <<No sé, pero las cosas siempre se han hecho así en este lugar>>.”*

Bastante educativo ¿cierto? El hecho es que esta fábula presenta la idea de paradigma en los monos, como un conjunto de ideas generado a través de la experiencia y posteriormente transmitido a los “nuevos integrantes” los cuales las replican sin cuestionar, a manera de dogma.

Así es como en TI existen diferentes enfoques o “paradigmas” para la resolución de problemas, como lo es el imperativo, el cual consiste en que se le indica a la máquina el qué va a hacer y cómo lo va a hacer, instrucción por instrucción, de manera secuencial. Este fue el paradigma representativo de las primeras generaciones de computadoras y los primeros lenguajes de programación, sin embargo, para los fines de este trabajo de investigación nos centraremos en un paradigma en particular: El paradigma orientado a objetos.

## **Paradigma orientado a objetos**

El paradigma orientado a objetos o OOP (por las iniciales en inglés de *Object-Oriented Programming*) se fundamenta en la visión de que todos los elementos que intervienen en un sistema se consideran objetos que pueden estar relacionados con otros objetos que se encuentren en el mismo sistema o en externos.

Estos objetos tienen características definidas, llamadas atributos, que los diferencian de otros objetos (por lo que cada objeto es único, por más atributos que comparta con otros objetos) Dichos objetos también pueden realizar acciones, éstas se conocen como métodos. Estos métodos pueden referirse a acciones que realiza el objeto en solitario o una comunicación con otros objetos.

Para que dicha comunicación se pueda llevar a cabo, los objetos se transmiten datos a través de mensajes, que pueden servir como entrada de datos para algún método o como un evento que desencadene una acción particular en el objeto. Existen diversas formas de representar y modelar sistemas orientados a objetos, la más utilizada es a través de modelos visuales centrados en mostrar a los objetos y sus relaciones con notaciones que faciliten su entendimiento, una de las herramientas más comunes es el Lenguaje de Modelado Unificado o UML (por las iniciales en inglés de Unified Modeling Language).

UML consiste en el lenguaje de modelado visual más utilizado por la industria de TI debido a que proporciona herramientas que permiten la representación detallada de los elementos que forman un sistema en todos los aspectos, incluido el de negocio en versiones recientes. Una de estas herramientas es el diagrama de clases, el cual permite modelar objetos para un sistema, sus relaciones y tipos de mensajes que compartirán, con el fin de delimitar los alcances de dicho sistema, que debido a su carácter visual funge como apoyo en el proceso de diseño y desarrollo.

## Bases de datos

Cuando hablamos de bases de datos ocurre un fenómeno curioso: Muchas disciplinas han incorporado el vocablo, pero cada una tiene diferentes aproximaciones al respecto. Por ejemplo, para las personas de Administración y Contaduría una base de datos es un archivo generado por una aplicación de *hoja de cálculo* como Excel en donde ellos vierten sus cifras, conceptos y datos, así como los controles que registrarán sobre los mismos. Y aunque no es erróneo, el concepto se vuelve restrictivo.

De manera general se denomina una base de datos al conjunto de datos almacenados en estructuras definidas que pueden relacionarse y que permiten su persistencia, manipulación y clasificación.

Existen varios tipos de bases de datos, pero las más utilizadas son las *relacionales*, las cuales consisten en estructuras llamadas tablas que permiten su manipulación a través de un sistema de columnas y filas, además de relacionarse entre sí a través de campos llave y utilizan el lenguaje de consulta SQL para la extracción de datos de dichas estructuras. Sin embargo, en los últimos años han tomado fuerza otros tipos de bases de datos llamadas NoSQL, las cuales se caracterizan por aceptar otro tipo de estructuras de almacenamiento para diversos tipos de datos no estructurados.

## Minería de datos

La minería de datos se refiere a técnicas y algoritmos mediante los cuales se puede extraer información de utilidad de volúmenes considerables de datos almacenados, de manera que se puedan “explotar” dichos datos y obtener información que permita ayudar a la toma de decisiones.

Es común que al hablar de minería de datos surjan otros conceptos como lo son los indicadores, que son información obtenida de la identificación de correlaciones entre variables y que permiten aportar información de gran utilidad sobre la situación actual de dichas variables e incluso permite generar proyecciones a futuro.

Actualmente la minería de datos ha tomado otro rumbo con la generación del llamado *Big Data*, el cual consiste en la limpieza, procesamiento y explotación de volúmenes inmensos de información, como la generada día a día por las personas que utilizan un servicio como las redes sociales.

## **Sistemas de Inteligencia de Negocios**

Estos sistemas son una de las evoluciones de la minería de datos, debido a que son sistemas expertos que permiten transformar los datos en información y ésta llevarla a conocimiento, optimizando el proceso de toma de decisiones.

Estos sistemas no solo se componen de medios de recolección, limpieza y procesamiento de datos. También incluyen algoritmos de explotación de dichos datos y mecanismos de predicción a través de la información generada. Son estas características por las que estos sistemas son enfocados a negocios.

## **Internet**

Un elemento que ha tomado una importancia considerable en el mundo actual, al punto de modificar los hábitos de las personas alrededor del mundo.

La red de redes se refiere al conjunto de infraestructura, protocolos y servicios que permiten la interconexión y transferencia de archivos entre los dispositivos a nivel mundial. Se le llama red de redes debido a que se trata del conjunto de redes interconectadas a través de diferentes tecnologías, la más utilizada es la familia de protocolos de comunicación *TCP/IP* o *Protocolo de control de transmisión/Protocolo de Internet*, los cuales permiten el enrutamiento de conjuntos de datos encapsulados, llamados paquetes, característica que simplifica el envío y recepción de datos entre equipos.

Es importante hacer énfasis en que la WWW (por las iniciales en inglés de *World Wide Web*) no es lo mismo que Internet: WWW se refiere a uno de los tantos servicios que componen a Internet, y que permite el intercambio de archivos de hipertexto (mejor conocidos como *páginas web*) y que en la actualidad representa su servicio más utilizado.

## Metodología OpenUp<sup>2</sup>

OpenUp consiste en una metodología de desarrollo de software, es decir, en un proceso que permite definir y supervisar las actividades necesarias para construir software que prioriza el trabajo colaborativo y la retroalimentación con el cliente con el fin de generar productos que otorguen valor a los mismos.

Creada por la comunidad Eclipse con el fin de proporcionar las herramientas necesarias para que equipos de trabajo pequeños pudieran desarrollar proyectos tecnológicos de múltiples tipos, es una iniciativa libre que toma las características de varios enfoques populares en el área y genera un enfoque fresco y llamativo.

Entre estas características destacan un enfoque iterativo e incremental en donde se prioriza el presentar entregables funcionales a los clientes con el fin de obtener una retroalimentación que permita el avance del desarrollo y al proceso mejorarse, reduciendo así el tiempo y esfuerzo de tareas que no son indispensables. Esta es la razón por la que OpenUp se define a sí mismo como un proceso que contiene los elementos mínimos necesarios para que un equipo de trabajo pueda comenzar a laborar, pues el proceso permite la construcción de software de calidad y escalable, ya que se puede utilizar como proceso base para el equipo de desarrollo, pero éste puede configurarse según las necesidades permitiendo la adición, modificación o sustracción de los elementos base.

---

<sup>2</sup> La información presentada en esta sección fue extraída de la documentación oficial, alojada en la página web oficial del proyecto <http://epf.eclipse.org/wikis/openup/>.

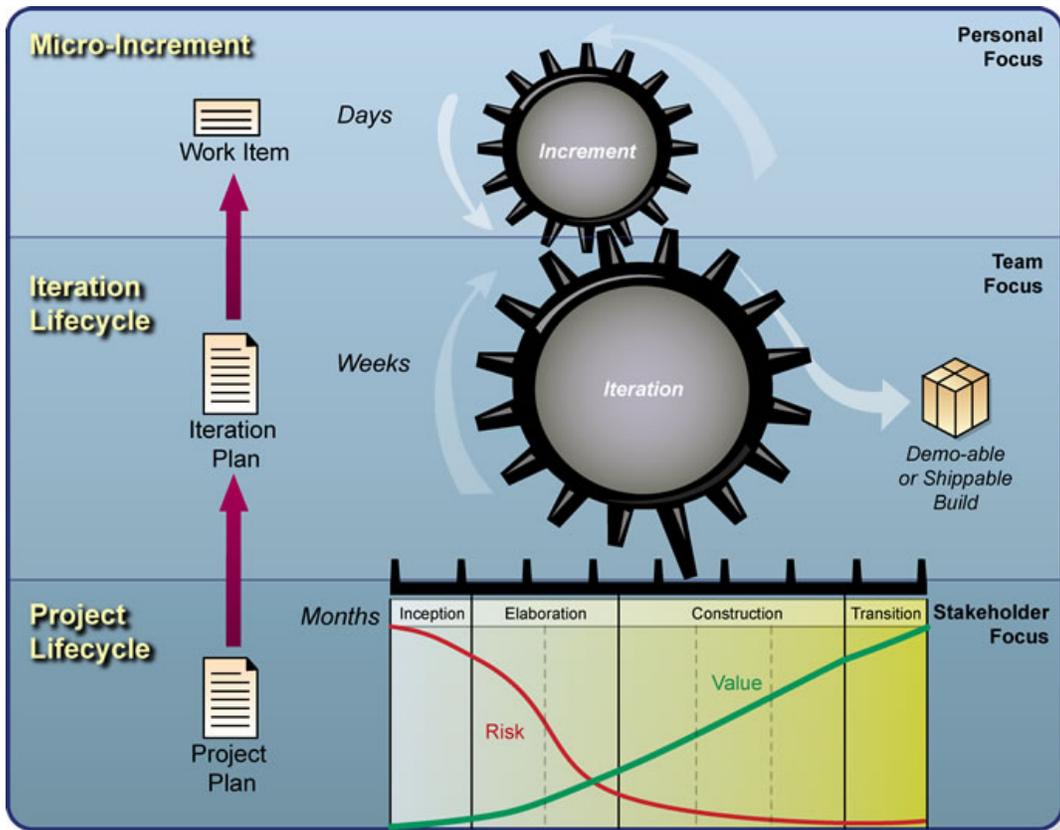


Ilustración 1: La filosofía de OpenUp se basa en como el proyecto es organizado en iteraciones compuestas por micro-incrementos, los cuales permiten el desarrollo de las etapas sin perder de vista la visión general con la que se planeó en un inicio (The Eclipse Foundation, 2016).

La metodología trabaja en dos grandes dimensiones: La primera es la dimensión del método, que se refiere a todos los elementos que intervienen en el desarrollo de software, independientemente del momento y el orden en que estos se utilicen. En esta dimensión se encuentran los siguientes elementos base:

- Roles: Los roles representan a los integrantes del equipo de desarrollo, los cuales intervienen en éste. Incluye algunos roles predefinidos listados a continuación, sin embargo, la metodología ofrece la libertad de poder agregar más según las necesidades y características del desarrollo:
  - Stakeholder: Representa al usuario final, no necesariamente el cliente, pero si quien manipulará el producto.
  - Analista: Especializado en las labores de levantamiento de requerimiento y análisis de necesidades.

- Arquitecto: Se enfoca en el diseño, implementación y validación de la arquitectura sobre la cual se construirá el desarrollo.
- Desarrollador: Encargado de la construcción del producto según la visión y especificaciones del analista y del arquitecto, con el fin de satisfacer las necesidades del cliente.
- Tester: Enfocado a las pruebas del software y también participa en las validaciones de otras tareas durante el proceso de desarrollo.
- Administrador de proyecto: Encargado de la definición, comunicación y supervisión del proceso de desarrollo durante la construcción del producto.
- Disciplinas: Para su proceso OpenUp se enfoca en los requerimientos, arquitectura, desarrollo, testing, administración de proyectos y la administración de configuraciones y cambios, como piezas fundamentales de un proyecto. Se ignoran otras disciplinas debido a que no son necesarias para equipos de trabajo pequeños o, en caso de necesitarse, su seguimiento debe llevarse de manera externa al desarrollo (Como en otras áreas, por ejemplo).
- Tareas: Las tareas son unidades mínimas de trabajo que se realizan por un rol específico, que se responsabilizará de la misma, y que tiene comunicación con otros roles que sirven de soporte durante el desarrollo de la actividad, interviniendo en alguno de los pasos de dicha tarea o como proveedores de información necesaria para la realización.
- Artefactos: Los artefactos son un elemento que se puede usar, modificar o ser el resultado de alguna tarea. Contienen información sobre el avance del desarrollo. Sin embargo, no necesariamente deben estar concentrados en presentaciones formales (Como un documento), pero son válidas en presentaciones como en reuniones recurrentes o en pizarrones.

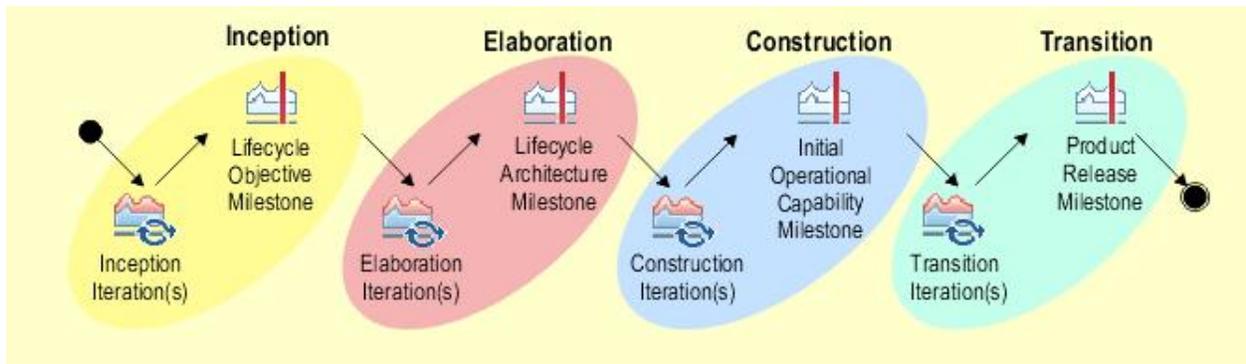
Por otro lado, se tiene la segunda dimensión del proceso. la que se encarga del ambiente temporal del desarrollo, esto es, del cómo y en qué momento se utilizaran los elementos de método, así como la organización del proyecto.

En esta dimensión tenemos dos grandes características:

- La primera consiste en el desarrollo de métodos reusables que especifican paso a paso la realización de una tarea con el fin de lograr una meta, tomando los elementos de la dimensión del método y organizándolos en secuencias semi-ordenadas específicas para un tipo de desarrollo. Estas secuencias son llamadas patrones de capacidad y son enfocados a un área en particular, por lo que forman los bloques básicos sobre los que se pueden construir patrones más complejos o procesos de desarrollo enfocados a cierto tipo de desarrollos.
- La segunda característica consiste en el ciclo de vida del desarrollo. La metodología OpenUp está basada en el proceso unificado: Metodología de desarrollo creada en 1998, adquirida en 2003 por IBM y de la cual se deriva un complemento que añadiría, entre otras cosas, estándares de documentación.

Las cuatro fases que componen el ciclo de vida de un desarrollo son:

- Fase de inceptión o inicio: En esta fase se engloban las tareas de análisis de la necesidad a resolver, por lo que se busca entender y delimitar dicha necesidad, identificar las funcionalidades más importantes para el producto a generar, así como definir los costos y riesgos asociados a dicha necesidad. Con estos elementos se puede formar la propuesta de una primera solución.
- Fase de elaboración: Se toma la información generada en la etapa de inceptión y se refina a modo de tener un mejor entendimiento del producto a crear. Con este refinamiento se pasa a diseñar y validar una arquitectura que obedezca a las especificaciones de la solución propuesta, reduciendo los riesgos de mayor impacto y que cumpla con los requerimientos refinados anteriormente.
- Fase de construcción: Consiste en la construcción iterativa e incremental de la solución definida en la fase anterior, con el fin de ofrecer un producto que cubra las necesidades definidas y esté listo para transferir a los usuarios.
- Fase de transición: En esta fase se hacen las pruebas con los usuarios con el fin de que estos validen las características del producto y su posterior entrega e implementación en el ambiente de producción.



*Ilustración 2: En la ilustración se muestra la sucesión de las fases que integran OpenUp, cada una de ellas por iteraciones en cada una de las fases y los hitos que se busca obtener con cada avance (The Eclipse Foundation, 2016).*

Esta forma de trabajo permite que el proceso de desarrollo se modifique de acuerdo a las necesidades, por ejemplo, un desarrollo en donde se necesite utilizar tecnología desconocida por el equipo de trabajo; entonces necesitará de algunas iteraciones extra en la fase de elaboración, durante el diseño y validación de la arquitectura del producto.

## Internet de las Cosas

Hablar de los descubrimientos e innovaciones que el hombre ha hecho a lo largo de la última década en el ámbito de tecnología no es solo hablar de cómo volver más eficiente un servicio o más pequeño un producto, engloba mucho más. Es la búsqueda constante de nuevas formas de comunicarse con su medio, de cómo poder conocerse a sí mismo a través del conocimiento de su entorno.

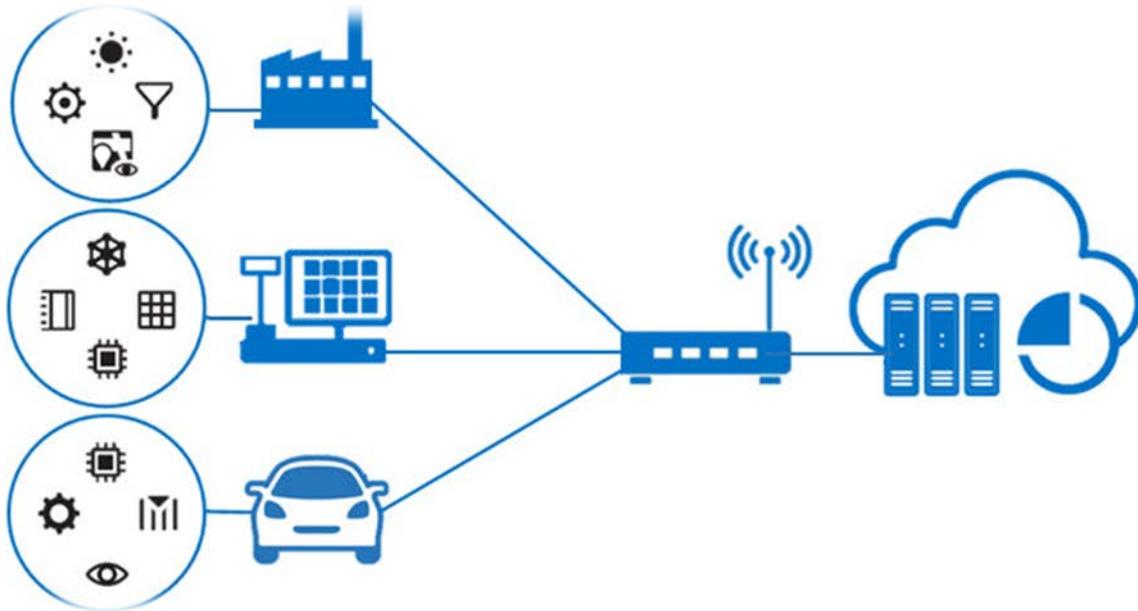
Con el nacimiento de Internet, el ser humano experimentó un fenómeno muy particular, nos referimos a que al dotarlo de la capacidad de compartir información con sus semejantes localizados en zonas geográficas completamente diferentes, las percepciones de distancia, de distancia-tiempo y fronteras comenzaron a verse alteradas: Las distancias se recortaban a micras de segundo, las fronteras iban desapareciendo poco a poco pues el contacto con personas en cualquier parte del globo estaba al alcance de un botón.

Gracias a este fenómeno, el ser humano ha ido adaptando su comportamiento y su entorno al avance tecnológico: La creación de las redes sociales, el computo en la nube o las herramientas colaborativas son ejemplos de cómo la tecnología está siendo encaminada hacia una dirección en particular. No solo busca interconectar al hombre, sino que lo desea volver más social, enfocándose en ofrecerle realizar sus actividades desde cualquier lugar y en el momento que desee. Y es que esta tendencia no solo va dirigida al hombre o a las computadoras, si no que poco a poco va incluyendo a más tipos de actores a dicha red.

Hablar de Internet de las cosas o IoT (por sus iniciales en inglés, Internet of Things) es hablar de un paradigma reciente, relativamente joven, que toma este enfoque de conectividad entre dispositivos (también llamadas cosas) y lo lleva a un nuevo nivel al incluir un factor tan común como lo es para nosotros el Internet. De manera general nos referimos a una red dentro de Internet dedicada exclusivamente a estas cosas que proveen datos sobre el ambiente en el que se encuentra el hombre y el cómo éstos intercambian información con otros sistemas como servicios en la red u otros hombres con el fin de ayudar a éste en su día a día.

Decir *cosas* puede sonar algo ambiguo e incluso absurdo, pero con el concepto se pretende abarcar un número infinito de dispositivos que permitan unirse con objetos de la vida cotidiana como sensores que permitan medir variables del ambiente donde se localizan, o actuadores que tengan una influencia específica en el entorno y que compartan una característica en particular: que son

dispositivos con una capacidad de cómputo limitada. Así, cuando hablamos de cosas en el Internet de las cosas no nos referimos exclusivamente a computadoras personales o laptops, tampoco a teléfonos inteligentes, si no a un conjunto de dispositivos con una funcionalidad particular que sea fácilmente acoplable a objetos tangibles y permita recopilar o influir de cierta manera en el medio.



*Ilustración 3: Esta ilustración muestra los elementos que conforman el IoT: Del lado izquierdo se tienen las “cosas” que censan el ambiente. Al centro se representa el acceso a Internet, medio por el cual envían y reciben datos. Del lado derecho se encuentra el almacenaje de los datos censados, los cuales suelen localizarse en la nube, es aquí mismo donde esos datos son explotados e interpretados, a modo de presentar información útil. (Intel, 2015).*

Es de esta forma en la que poco a poco comenzamos a dotar de cierta inteligencia a los objetos que nos rodean: Imagine usted que se dirige a trabajar, y antes de salir de casa recibe un mensaje en su teléfono inteligente (O algún dispositivo basado en tecnología wereable<sup>3</sup>) de parte del paraguas recordándole no dejarlo porque por la tarde lloverá, aproximadamente a la hora de su salida. O del refrigerador que le recuerda el comprar leche.

Lo anterior puede sonar exagerado y causar risa, como esas viejas visiones de los escritores de ciencia ficción durante la década de los 90’s sobre “El futuro y el nuevo milenio” con automóviles

---

<sup>3</sup> Se refiere a todo dispositivo que puede ser portado o adaptarse a alguna parte de un ser vivo. La Real Academia Española ha propuesto como traducción “Tecnología Ponible”, pero aún se discuten posibles traducciones.

voladores y colonias espaciales a lo largo del cosmos, pero a diferencia de estas historias de ficción, el Internet de las cosas es una realidad.

Es así como en lugar de tener pocos dispositivos con un gran poder de procesamiento (como pueden ser las computadoras personales o portátiles) se propone el tener muchos dispositivos con poco poder de procesamiento pero que gracias a su tamaño pequeño permiten implementarlos dentro de una cantidad infinita de objetos cotidianos, dotándolos de nuevas funcionalidades a bajo costo.

Esta interconexión de dispositivos o cosas, para utilizar el argot adecuado, utilizan Internet como plataforma de comunicación: Un sensor de temperatura puede avisarle a un ventilador que la habitación está a alta temperatura y es momento de encenderse, o también puede ser que el hombre pueda acceder a través de Internet desde su oficina u otro lugar y activar dicho ventilador.

Aunque en la actualidad esta interconexión no solo se limita a Internet, ya que se han ido sumando una variedad de protocolos y métodos de conectividad compatibles que buscan complementar y enriquecer el paradigma, la presencia de Internet en un ambiente de IoT es un elemento necesario.

El IoT representa uno de los grandes avances del hombre en lo que va de la década, pues no solo representa un paradigma novedoso que nos conecta más a nuestro ambiente al intentar echar abajo las fronteras del cómputo tradicional dominado por las computadoras personales, si no que representa una nueva visión del mundo al ampliar el panorama de trabajo a rincones que antes no hubiéramos considerado.

En los ambientes IoT pueden identificarse 3 bloques principales que lo conforman:

- En primer lugar, tenemos los objetos que nos interesa medir, estos pueden ser tangibles como una mesa, o una habitación, e intangibles como puede ser el viento, la temperatura, etc. Es importante definir de manera concreta dos cuestiones sobre los objetos a medir debido a las limitantes de los dispositivos a utilizar: La primera es qué tipo de característica o manifestación nos interesa y la segunda es qué tipo de interacción nos gustaría tener con dicha característica en el ambiente.
- En segundo lugar, una vez identificado lo que necesitamos y sobre qué lo necesitamos, requerimos buscar que tipo de dispositivos se utilizaran para dicho propósito, es decir, si censaremos exclusivamente el ambiente o si podremos tener una influencia en él a través de actuadores físicos.
- Finalmente nos queda la forma en la cual dichos dispositivos compartirán datos, ya sea que exclusivamente estén conectados a Internet o bien que existan otras tecnologías que permitan complementar el ambiente generado.

Estos bloques están siempre presentes en ecosistemas generados con una visión IoT. Sin embargo, esto no significa que puedan ser los únicos bloques existentes, al contrario, dicho paradigma es flexible y adaptable, por lo que se puede complementar y mejorarse según la situación (Y no dudo que en algunos meses esto ocurra).

## La estructura funcional del IoT

Adrian McEwen en su libro "Designing the Internet of Things" propone lo que él llama la ecuación del internet de las cosas<sup>4</sup>, que puede definirse como una relación funcional que consiste en la suma de objetos físicos, más actuadores, controladores y sensores más Internet. Esta ecuación es similar a los actores mencionados en la sección anterior, pero representa una clasificación propuesta sobre un modelo reciente y es por esto que me permito tomarla como referencia.

$$\begin{array}{c} \textit{Physical Object} \\ + \\ \textit{Controller, Sensor, and Actuators} \\ + \\ \textit{Internet} \\ = \\ \textit{Internet of Things} \end{array}$$

Ilustración 4: La "Ecuación del internet de las Cosas" (Adrian McEwen, 2013).

Actualmente las soluciones con características IoT son productos enfocados a solucionar necesidades muy particulares por parte de quienes los construyen y generalmente están dirigidos al ámbito del hogar: Regular la temperatura del inmueble, los consumos de servicios, facilitar ciertas tareas como la apertura del garaje o puertas de algunos espacios son aplicaciones muy comunes y que los propios creadores comparten para que la comunidad en línea comente y mejore con su experiencia, sin embargo estas soluciones comparten un detalle: Se limitan exclusivamente a la aplicación de conocimientos con el fin de solucionar un problema.

Esto no debería ser así pues otras áreas de conocimiento podrían beneficiarse de dichas soluciones si se les diera el enfoque adecuado y es una cuestión curiosa debido a que desde los inicios de la computación con la creación de la pascalina y las calculadoras mecánicas, pasando por la maquina analítica y las gigantescas primeras computadoras de bulbos, hasta los circuitos integrados creados con técnicas VLSI (Integración a muy grande escala), a pesar de haber beneficiado a los sectores de la investigación y las ciencias, las primeras aplicaciones reales de dichas tecnologías fueron dirigidas a los negocios: Calculo de nóminas e impuestos. Si bien es cierto que quien no conoce la historia está

---

<sup>4</sup> Aunque la definición formal de ecuación implica la existencia de al menos una variable, para fines del este trabajo se refiere a la igualdad o relación existente entre dos o más objetos.

condenado a repetirla, parece que en el ámbito de la tecnología esta frase se aplica de diferente manera, día a día las empresas mexicanas se han vuelto más resistentes al cambio y las nuevas tecnologías al rezagarse en sistemas de décadas pasadas que caen en la obsolescencia a una velocidad sorprendente. Ese es un nicho de mercado potencial que se está perdiendo, pues todas las tecnologías pueden aplicarse a los negocios y generarles un valor y con ello, una ventaja competitiva.

## **Modelo Machine-to-Machine (M2M)**

El origen del paradigma IoT está basado en un modelo ya existente llamado M2M el cual consiste en la interconexión de dos o más dispositivos con un fin definido, por ejemplo, él envió de datos entre nodos cercanos, realizar acciones automatizadas determinadas, etc.

Algunos de los productos surgidos eran para uso cotidiano, como las lámparas que se encienden y apagan de manera automática al entrar o salir de una habitación, que consiste en un sensor infrarrojo o de movimiento que tiene comunicación con la luminaria. Por otro lado, las aplicaciones para el sector empresarial abarcaban medidores de consumo como los de agua o electricidad, el seguimiento en transportes de carga de manera remota o la telemetría de entregas, los cuales vieron modificadas su proceso de obtención al incluir hardware especializado como PDA's o localizadores satelitales, incluso la construcción de edificios inteligentes (Torres ejecutivas, centros comerciales, instituciones de gobierno y educación, etc.).

Sin embargo, la comunicación de los dispositivos se volvía complicada debido a la cantidad de protocolos de comunicación existentes. Es así como algunos usaban radiofrecuencias, o la red satelital, o en algunos casos, la descarga de estos datos hacia los sistemas debía ser manual. En el ambiente comercial, estas interacciones se limitaban al área donde se encontrarán: Las luces de una habitación, sensores que abrían o cerraban puertas, ya fuera a través de controles manuales o de sensores de corto alcance.



*Ilustración 5. Algunas aplicaciones que surgieron con la aplicación del M2M a el sector empresarial.*

Al tomar los beneficios del modelo M2M y enriquecerlo con un enfoque de acceso ubicuo (El principio en el que se basa el cómputo en la nube, que consiste en asegurar el acceso a uno o varios servicios sin importar el lugar geográfico donde se encuentre) se proporciona un medio de comunicación común, estandarizado y barato. Esta característica detono una evolución, que desemboco en lo que ahora conocemos como IoT.

### **Actuales plataformas que apoyan proyectos IoT**

Aunque existen muchas iniciativas que apoyan el IoT, son principalmente dos plataformas libres las que apoyan el desarrollo de proyectos y prototipos: Arduino y Raspberry Pi. Su éxito ha llamado la atención del mercado y en los últimos años han surgido una serie de alternativas que tienen como fin apoyar al desarrollo de dichas plataformas al proveer de servicios que faciliten a los desarrolladores la generación de soluciones.

### **Plataformas en la nube**

Debido a que los dispositivos generan una cantidad considerable de datos a lo largo del tiempo en que están activos, generalmente se necesita un lugar donde almacenarlos con el fin de procesar, visualizar o trabajar con ellos.

Existen una cantidad importante de sitios que permiten almacenar datos en sus nubes privadas, con el fin de evitar los costos que se podrían derivar de la contratación, mantenimiento o configuración de una base de datos por parte de los desarrolladores, delegando dichas actividades a los propietarios del servicio.

Han surgido una serie de servicios por parte de empresas que permiten trabajar con dichos datos alojados en la nube sin necesidad de descargar software adicional al tratarse de peticiones entre servicios web. Un ejemplo de estos servicios es la Google App Engine, una plataforma que permite englobar los servicios de diferentes productos del ecosistema Google como lo serian el uso de Drive para la creación de documentos en la nube y su vinculación con orígenes de datos, Graphs para la visualización de dichos datos a través de gráficos y elementos visuales vía web.

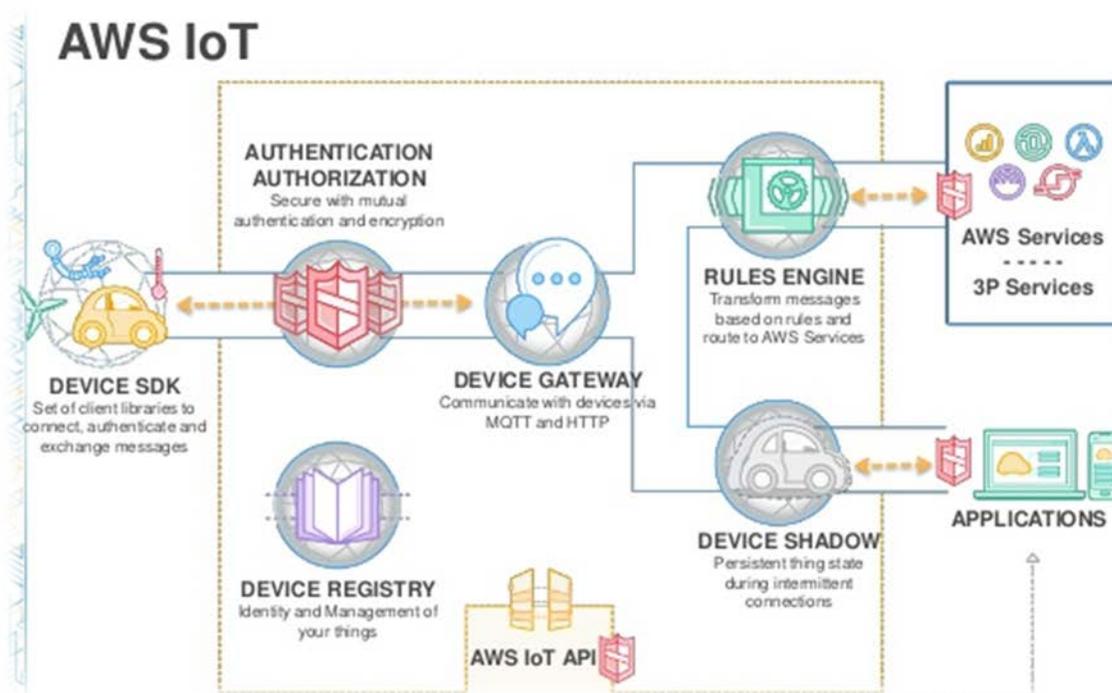


Ilustración 6. El flujo de trabajo de la aplicación de IoT desarrollada por Amazon, así como la interacción con el resto de los servicios de su plataforma Cloud AWS. (Amazon, 2015)

## **Windows 10 IoT**

Microsoft también ha anunciado una versión de la próxima actualización de su sistema operativo insignia enfocado a IoT en exclusiva para la Raspberry Pi 2, el cual estaría optimizado para dicha plataforma y ofreciendo las herramientas que actualmente provee la versión de Linux nativo, pero añadiendo una interfaz transparente al ecosistema Windows del resto de dispositivos compatibles con dicho sistema operativo. El propósito es simplificar el desarrollo de aplicaciones que permitan la interconexión de la plataforma utilizando Windows 10 como entorno.

Adicionalmente Microsoft también ha lanzado su propuesta para los usuarios de Arduino al implementar las funciones de comunicación entre las placas y el sistema operativo de manera nativa en la consola nativa PowerShell, de esta manera se podrían hacer las funciones de compilación, carga de código y visor serial del IDE oficial Arduino nativamente en Windows.

La publicación de este apoyo por parte de una empresa tan importante en la industria como lo es Microsoft no debe tomarse a la ligera, pues al proporcionarlo de manera directa a las plataformas IoT, Microsoft solo confirma la potencia del uso de estas tecnologías y la importancia que adquirirán en un futuro muy cercano.



## Capítulo 2

### DISPOSITIVOS EN EL HARDWARE LIBRE

#### ¿Qué es el hardware?

Según el diccionario de la Real Academia Española (RAE) se define al hardware como “*el conjunto de los componentes que integran la parte material de una computadora.*”. Hasta hace unos años el tema del hardware era completamente privativo, empresas como IBM o Intel ejercían una omnipresencia en los equipos de cómputo durante finales de los 90’s y principios del 2000, reservando el enfoque libre a computadores caseros, compuestas principalmente de compuertas eléctricas y pequeños circuitos construidos a mano, pues imprimir placas con circuitos con mayor complejidad era caro.

Esto ha cambiado con el paso del tiempo, al grado de poder tener una pequeña computadora del tamaño de una tarjeta de crédito a un costo muy bajo (\$500.00 M.N. promedio) capaz de ejecutar una versión de Linux y realizar diversas tareas. La presencia de estos componentes físicos de bajo costo y con un alto potencial de procesamiento han hecho que en los últimos años surjan algunos fenómenos como el internet de las cosas, tema que ya se ha mencionado brevemente y en el que se profundizará en capítulos posteriores en este trabajo y en el que empresas como Cisco, Google, Microsoft, Apple entre otras, han comenzado a presentar sus propuestas para abarcar este mercado en crecimiento.

Cuando hablamos de hardware libre, a diferencia del software, nos referimos a que la información referente a cierto dispositivo se encuentra disponible para el público e incluye los códigos y lenguajes que éste entiende, información sobre su composición (que puede especificarse a través de diagramas eléctricos o planos de impresión en los que se explique la distribución de los circuitos que lo componen), características eléctricas de su funcionamiento (agrupadas en lo que se conoce como datasheets u hojas de datos del dispositivo), las funciones o mecanismos con los que trabaja, etc.

Esto con el fin no solo de poder, siguiendo la información distribuida, generar una copia del dispositivo, sino también mejorarlo o construir alguna interfaz o dispositivo compatible con él, dotándolo de nuevas funcionalidades.

Un ejemplo de estas prácticas son la gama de microcontroladores Arduino (de los que hablaremos a profundidad más adelante) los cuales proporcionan la facilidad de poder modificar los diseños eléctricos de sus placas e imprimir las propias y que además ofrecen la opción de generarlas. Con esto muchas empresas no solo han copiado los modelos y revendido como propios, sino que además se han generado una cantidad grande de placas externas que dotan al microcontrolador de nuevas funcionalidades como conectividad a internet, interfaces para ciertos dispositivos, y otras más

## **Iniciativas open source o libres**

Utilizar tecnologías libres tiene muchos beneficios en comparación con el uso de marcas privativas, puesto que explotan los beneficios del enfoque open source y con esto generan un valor para sus usuarios. Primeramente se puede hablar acerca de costos, debido a que la licencia necesaria para el uso de software libre es, en la mayoría de los casos, gratuita y permite ahorrar los cuantos miles de pesos que son destinados a la adquisición de licencias de uso para software privativo, un ejemplo de esto es la aplicación para edición de fotografía de Adobe, Photoshop, que en su modalidad de servicio puede costar entre 20 y 30 dólares al mes, mientras que aplicaciones como Gimp que es software libre y ofrece las mismas herramientas que Photoshop son completamente gratuitas. Esto quizás no sea representativo si hablamos de una sola licencia o plan de licenciamiento, pero si agregamos que el costo por año de un solo producto puede elevarse hasta los 160 dólares anuales por dispositivo que haga uso del servicio, el ahorro comienza a ser significativo.

No nos quedemos solo con el ambiente del software, en el hardware también existen casos similares. Si habláramos de sensores que permitan medir algunas variables en el agua, como puede ser el pH o la temperatura, podemos citar que una probeta de precisión industrial como las que distribuye una marca privada pueden costar entre \$256 y \$520 dólares solamente la probeta, mientras que un kit de hardware abierto que pueda competir con la precisión de esta probeta y que además de la probeta incluya un traductor y conector, además de soluciones para calibración y acceso a la documentación puede costarse en poco menos de \$200 dólares. Este tipo de comparativas son las que poco a poco nos hacen ver por qué el enfoque libre se ha fortalecido en los últimos años.

Otro de los beneficios proviene de la retroalimentación y mejora que pueden proporcionar las distintas comunidades, foros y sitios de reunión especializados en el tema que existen en internet,

ya que esto nos permite asegurar que un producto está orientado al usuario final, y que puede representar un proceso de mejora continua.

Sin embargo, estos beneficios pueden actuar en función del tipo de necesidades que tenga el usuario, ¿A qué me refiero con esto? Si bien es cierto que este tipo de software recibe un enorme número de actualizaciones en pequeños lapsos, esto no sería del todo benéfico para sistemas que fueran críticos, por ejemplo, no se podría estar parando para una actualización sistemas transaccionales que ejecuten tareas críticas. Es por eso que, a pesar de las ventajas que puedan ofrecer las tecnologías libres, también es cierto que se debe hacer un análisis detallado de las necesidades del usuario para saber en qué rubros puede implementarse una tecnología libre y en cuales más conveniente alguna privativa.

## **Diferencia entre microcontroladores y microprocesadores**

Es importante comentar que existen muchos tipos de dispositivos libres como sensores de diversas variables, actuadores para cantidad de acciones, motores que permitan interacción con otros objetos, etc. Pero para poder interactuar con estos dispositivos, independientemente del objetivo que se quiera lograr, es necesaria una controladora de algún tipo.

La primera interacción con estos dispositivos se dio a través del uso de botones y switches interconectados por protoboards: rectángulos de cobre debajo de una cuadrícula de plástico en donde se construyen conexiones utilizando cables de cobre y que permiten llevar un orden, así como facilitar el trabajo de conectar dispositivos. Sin embargo, era una manera muy básica y rústica de generar circuitos.

Después se comenzaron a utilizar las de placas de circuitos impresos (también llamadas simplemente PCB) las cuales consisten en placas no conductoras sobre las que se imprimen pistas de cobre que sirven de “camino” entre los componentes, con las que se pueden comunicar entre ellos sin necesidad de cables. Conforme el proceso de construcción fue madurando, el tamaño del producto final se fue reduciendo. Esto representó un gran avance pues a través de PCBs se podía generar un circuito complejo que integrara múltiples componentes, de manera sencilla.

Tiempo después surgieron los microcontroladores, que son componentes que permiten entender y realizar tareas, que se programan y “graban” en su memoria interna, con el propósito de que ejecute

acciones. Tiene la ventaja de que facilita el manejo de datos a muy bajo nivel (bytes) debido a que sus instrucciones permiten un mejor manejo de la memoria escasa con la que cuentan.

Por otro lado, existen los microprocesadores, concepto que es muy similar al de un procesador montado en una motherboard. El microprocesador es una unidad que incluye módulos que permiten el uso de un mayor número de funciones y de una complejidad mayor, sacrificando una parte de la libertad con la que se puede manipular la memoria interna en pro de la capacidad de procesamiento.

# Arduino

## Historia

La historia de Arduino nos remonta al año 2005 en Italia, con Massimo Banzi, estudiante del Interaction Design Institute Ivrea, localizado en el pueblo de Ivrea, al norte de Italia. La idea de crear una placa microcontroladora de bajo costo surge debido a que aun en 2005 los precios de éstas eran muy altos para un estudiante además de que las existencias del instituto eran pocas y un tanto obsoletas. Así es como Banzi pretendía apoyar a sus compañeros durante su formación académica y ayudar financieramente a la escuela debido a que estaba a punto de quebrar y, por ende, de cerrar.

El prototipo inicial fue creado dentro del instituto: Consistía en una simple placa donde se colocaba el microcontrolador habilitado a sus resistencias y una interfaz un tanto simple a la que solo podían conectarse componentes simples como leds. Algunos años más tarde, el estudiante colombiano Hernando Barragán, tras enterarse del proyecto Arduino se une al equipo de Banzi con el desarrollo del primer entorno de programación, llamado Wiring que está basado en el lenguaje de programación *Processing*, y fue junto a David Mellis (Estudiante de Ivrea) con quien mejoraron el entorno de programación.

Al equipo de Banzi se unieron otras personas quienes aportaron mejoras para su placa, tales fueron los casos de David Cuartielles quien ayudo a la mejora de las interfaces para la interacción con el hardware y de Tom Igoe quien tras conocer el proyecto contribuyó con el soporte para USB nativo de la placa, él fue la primera persona en sugerir la distribución mundial de la placa.

Cuando el prototipo estuvo completo y tras pasar algunas pruebas, el producto comenzó su distribución en el campus de Ivrea. En este momento es cuando aparece Gianluca Martino, publicista que apoyó con la distribución local del producto, que tuvo una aceptación total entre los estudiantes, profesores y conocidos, lo que fue el aliciente final para que Banzi se convenciera de producirlo en masa y distribuirlo por toda Italia.

Este fue el inicio de la empresa que conocemos, en el año 2015. Arduino no solo ha fortalecido su presencia a nivel mundial, también ha obtenido el apoyo de fabricantes de componentes y dispositivos eléctricos quienes han generado sus productos compatibles con el famoso microcontrolador, también han logrado diversificar sus productos al incluir placas especializadas

para conectividad a internet, microcontroladores, microprocesadores y procesadores más poderosos (como el Edison de Intel), wearables e incluso impresión 3D.

## **Principales Productos y Características**

Aunque la marca Arduino ha ido evolucionando e incursionando en diferentes mercados con una variedad de productos, los más destacados son los microcontroladores, los cuales tienen unas características muy similares entre sí. Entre sus productos más famosos se encuentra Arduino Uno, una placa sencilla y de fácil manejo, con la que comúnmente se recomienda iniciar en el mundo de los microcontroladores, también existen productos más sofisticados como Arduino Yún, un microcontrolador que incluye un microprocesador ejecutando una versión de Linux, que permite interacciones más complejas. Entre las características que estos nos ofrecen se encuentran las siguientes.

### ***Microcontrolador***

Principalmente hablamos de un microcontrolador de la marca ATmega (cuyas características y esquemas se pueden consultar a través de la página oficial de la empresa) que cuenta con las interfaces necesarias para trabajar con los diferentes componentes de la placa. Incluye 1Kb de almacenaje interno para códigos compilados y 2Kb de SRAM para variables, funciones, manejo de datos u otro tipo de interacción que se necesite programar.

Si bien estamos hablando de recursos muy limitados se debe a que la operación se realiza a un nivel de abstracción bajo, manipulando localidades de memoria, señales digitales y análogas, bytes, etc. por lo que el uso de técnicas de diseño y optimización de algoritmos son de gran utilidad para poder aprovechar al máximo las características de la placa.

## ***Alimentación***

La alimentación se da por medio del puerto USB, aunque algunas placas como la UNO incluyen un puerto dedicado a alimentación externa, que puede ser de 6v o 12v <sup>5</sup>, con lo cual la placa puede trabajar sin presentar ninguna inestabilidad.

Las placas contienen dos tipos de pins referentes a la alimentación eléctrica: los pins de 5V y de tierra (también identificada bajo la abreviatura GND), a partir de ellos se realiza la distribución de energía a los sensores y actuadores que se utilicen. También existen otros pines como el de 3.3V y el pin Vin que puede proveer corriente directamente a la tarjeta si así fuera requerido.

## ***Memoria***

Como se comentó antes, la memoria de estas placas es reducida, pero gracias al uso de módulos extra es posible añadir almacenamiento externo a través del uso de memorias, principalmente del tipo SD (Por sus siglas en inglés Secure Digital) dada su bajo precio y alta popularidad entre los teléfonos inteligentes, permitiendo expandir la capacidad de almacenaje desde 1 Gb hasta 32Gb o más, para otros modelos.

## ***Conexiones de entrada y salida***

Las placas Arduino cuentan generalmente con 6 Pins dedicados a dispositivos analógicos y 13 para digitales, sin embargo, hay que tener en consideración que algunos de estos pins se requieren para realizar ciertas acciones, como leer o escribir en memoria, realizar envío o recepción de datos a través de internet, etc. Por lo que este número se ve severamente reducido. Hay placas como la Arduino Mega que contienen un mayor número de pins disponible para prototipos que necesiten un número mayor de pins.

---

<sup>5</sup> La experiencia al llevar a cabo las pruebas de concepto presentadas en el capítulo 5 nos demostraron que las placas trabajan bien con una alimentación de 5V, por lo que un adaptador de pared utilizado por teléfonos inteligentes puede utilizarse sin causar problema alguno.

## Comunicación

Por defecto la mayoría de placas solo cuentan con un puerto USB que sirve también como una interfaz serial entre la PC y la placa. Según las necesidades del prototipo actualmente existen diferentes placas complementarias que permiten la conexión a internet de manera alámbrica, inalámbrica e incluso celular, por no mencionar otros protocolos como el Bluetooth, o las radiofrecuencias.

Por otro lado, para poder realizar la programación de estas placas se hace uso del entorno de desarrollo oficial de Arduino, el cual se escribe en lenguaje Wiring, un lenguaje estructurado muy similar al lenguaje C, que también puede utilizarse para la generación de programas. Como nota, cabe resaltar que la placa soporta programación orientada a objetos a través de la sintaxis de C++, pero es necesario hacer énfasis en que debe existir un manejo de memoria impecable si queremos que el uso de este paradigma no colapse al microcontrolador.

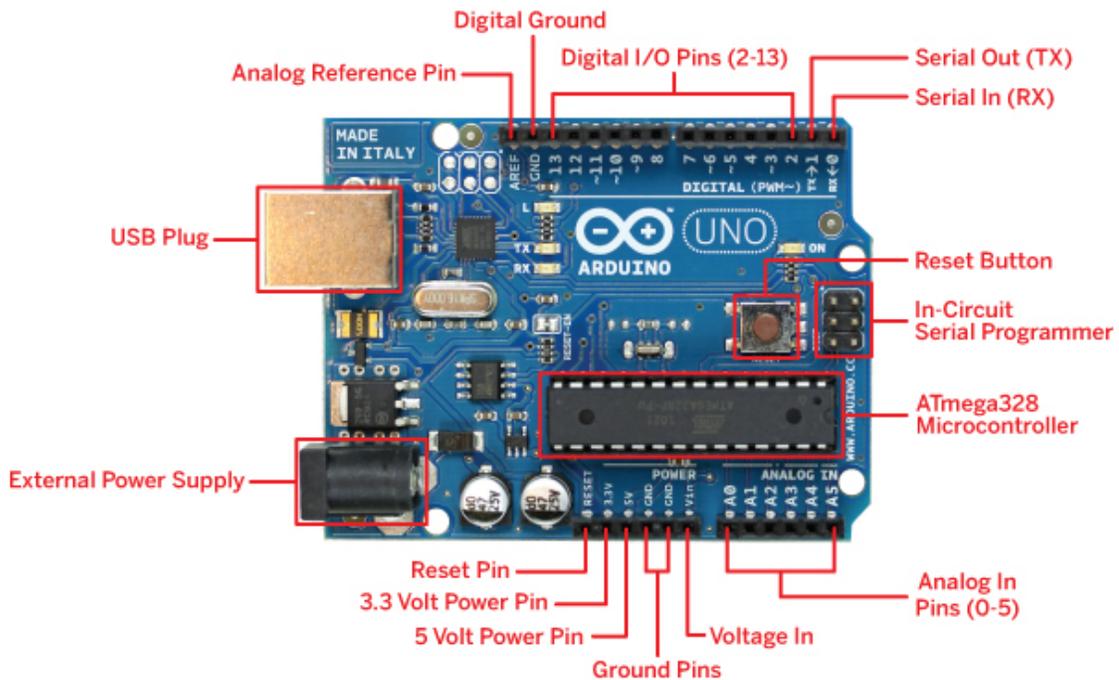


Ilustración 8: Un diagrama que muestra los componentes del microcontrolador Arduino UNO.

## Raspberry pi

El microprocesador Raspberry Pi se define a sí mismo como *Una computadora de bajo costo y del tamaño de una tarjeta de crédito*. Fue desarrollado por la fundación del mismo nombre en el año 2011 en la Universidad de Cambridge, Reino Unido, con el objetivo de apoyar a la enseñanza de la computación en las escuelas y a la niñez en general. Su comercialización se dio hasta un año después.

En esencia el Raspberry Pi contiene los componentes básicos necesarios para el funcionamiento de una computadora que ejecuta como sistema operativo una distribución Linux. Entre las principales características técnicas del producto, este contiene:

- Un chipset, generalmente de la marca Broadcom, junto a un microprocesador a 700 MHz que podría ser a 1GHz sin comprometer el equipo (aproximadamente la velocidad a la que llegaba el Pentium III en la década de los 90's).
- Un procesador gráfico, así como un puerto HDMI que permite una conexión con monitores externos.
- 512Mb de memoria RAM
- Un conector RJ45 que proporciona acceso alámbrico a internet de hasta 10/100 Mbps.
- Conectores USB
- Jack 3.5mm para salida de audio
- Pines de entrada y salida de propósito general
- Lector SD

Como se puede apreciar la Raspberry Pi representó toda una innovación que dio como origen múltiples proyectos interesantes y que aun ahora sigue sorprendiendo: Consolas de videojuegos basados en ella, iniciativas de aprendizaje en países de bajos recursos, incluso como herramienta para programas de penetración y hackeo.

## **Historia**

Su historia comienza en el año 2006, mismo año en que Massimo Banzi inicia su proyecto Arduino. En este año se genera el primer prototipo de Raspberry Pi y su diseño y esquemas se publican en internet para su descarga pública.

En mayo del 2009 fue creada la Fundación Raspberry Pi en South Cambridgeshire, Reino Unido, con la intención de llevar el uso y entendimiento de las computadoras a los niños desde edades muy tempranas, de manera que puedan conocer su funcionamiento interno y se involucren en el mismo. La idea vino de la mano de un desarrollador de videojuegos, David Braven, se pretendía que una vez que los niños entendieran cómo funcionan las computadoras y sus interfaces, pudieran ellos mismos generar sus componentes de las maneras más creativas que surgieran y, sobre todo, de forma divertida.

El primer prototipo llegó en el año 2011 con el nombre de Modelo A, la primera forma de adquisición de dicho modelo fue a través de una subasta en eBay, venta que fue todo un éxito y con lo recaudado se pudo apoyar el desarrollo de una siguiente versión más estable.

El Modelo B contaba con el doble de RAM de su predecesor, así como un puerto para conexiones alámbricas a internet, y fue mandado a producir directamente a un fabricante chino, permitiendo ofrecer una calidad mayor que su modelo anterior a un precio menor, lo cual marco el inicio de un éxito de ventas casi instantáneo, llegando incluso a saturar los servidores de la fundación durante la fecha de lanzamiento al público general.

Actualmente Raspberry Pi ha lanzado la preventa de su modelo 2, el cual cuenta con el apoyo de algunas compañías como Microsoft, quien desarrolló una versión de su nuevo sistema operativo Windows 10 en exclusiva para el Raspberry Pi 2 llamado Windows IoT.

## **Principales Productos y Características**

Raspberry Pi cuenta con dos productos insignia: El Modelo A y B, los cuales se diferencian por la cantidad de RAM incluida 256 Mb y 512 Mb respectivamente, además de que el modelo A no incluye un puerto RJ45 por lo que su costo es menor. Con respecto al resto de las especificaciones de los productos siguen la misma composición que es la siguiente.

## ***Procesador***

El Procesador que incluye el Raspberry Pi es un Broadcom que tiene tecnología SoC (*System-On-Chip*, *sistema dentro del chip* en español) esto significa que la mayor parte de los componentes del ecosistema se encuentran embebidos dentro de un único chip en la placa, esto incluye el CPU, GPU y el chipset, así mismo se diferencia de un procesador X86 común en pc's debido a que utiliza una arquitectura ARM (Advanced RISC Machine), caracterizada por su bajo costo, un reducido número de transistores para su construcción y bajo consumo eléctrico, lo que los hace ideales para equipos cuya principal fuente de energía sean las baterías.

## ***GPU***

El procesador gráfico incluido permite reproducir contenido en Blue-ray utilizando la codificación libre estándar, así como desplegar gráficos en 1080p a 30 fps, lo cual lo dota de un potencial gráfico enorme en un espacio tan pequeño.

## ***Memoria Interna***

Como lo mencionamos en su momento el Modelo B es quien tiene una memoria RAM del tipo SRAM de 512Mb.

## ***Almacenamiento***

Debido a las limitaciones de tamaño, ninguno de los modelos contiene un disco duro o algún medio de almacenamiento interno. Sin embargo, se cuenta con un lector de tarjetas SD cuya única condición es que estas sean de al menos 2Gb de capacidad, pues en esta residirá el sistema operativo, el cual debe ser "instalado" previamente antes de poder ser ejecutado por la Raspberry Pi.

## ***Salidas de Audio y Video***

La placa cuenta con un Jack estándar que permite la conexión a bocinas, audífonos y otros medios de reproducción compatibles con el Jack de 3.2mm, un estándar ya en el mercado

Por otro lado, para la salida de video se cuenta con tres opciones: La primera es el clásico conector RCA que permite llevar video y sonido a dispositivos antiguos como televisores CRT. Otra opción

viene de la mano de un puerto HDMI con video + audio por el mismo canal, lo que le da una compatibilidad con las actuales televisiones de pantalla plana y calidades superiores de imagen. Finalmente, también contiene una salida DSI (Display Serial Input) para paneles LCD de 7", los cuales los propios distribuidores del producto suelen comercializar junto con la placa.

### ***Conexión de entrada y salida***

Para la interacción con el ambiente externo el Raspberry pi cuenta con tres mecanismos principales: Puertos USB que permiten conectar pendrives y otros dispositivos de almacenamiento como discos duros portátiles, celulares, etc.

Una tarjeta de red (Solo el modelo B) que permite la conexión a internet e incluso la conectividad con una pc a través del estándar de cableado correspondiente, lo cual abre las posibilidades de aplicación del producto.

Finalmente incluye un total de 8 pines GPIO, los cuales son solo interfaces, ya sea de entrada o de salida, que son completamente programables y permiten la interacción con otro tipo de dispositivos como sensores o actuadores.

### ***Alimentación***

La placa se alimenta a través del puerto micro USB que viene incluido, éste proporciona un voltaje de 5V, suficiente para su encendido y utilización, pues carece de un botón de encendido/apagado.

Este voltaje lo proporcionan la mayoría de los cargadores de teléfonos inteligentes que existen en el mercado, por lo que el suministro de energía pasa a ser de baja prioridad para el usuario de la placa.

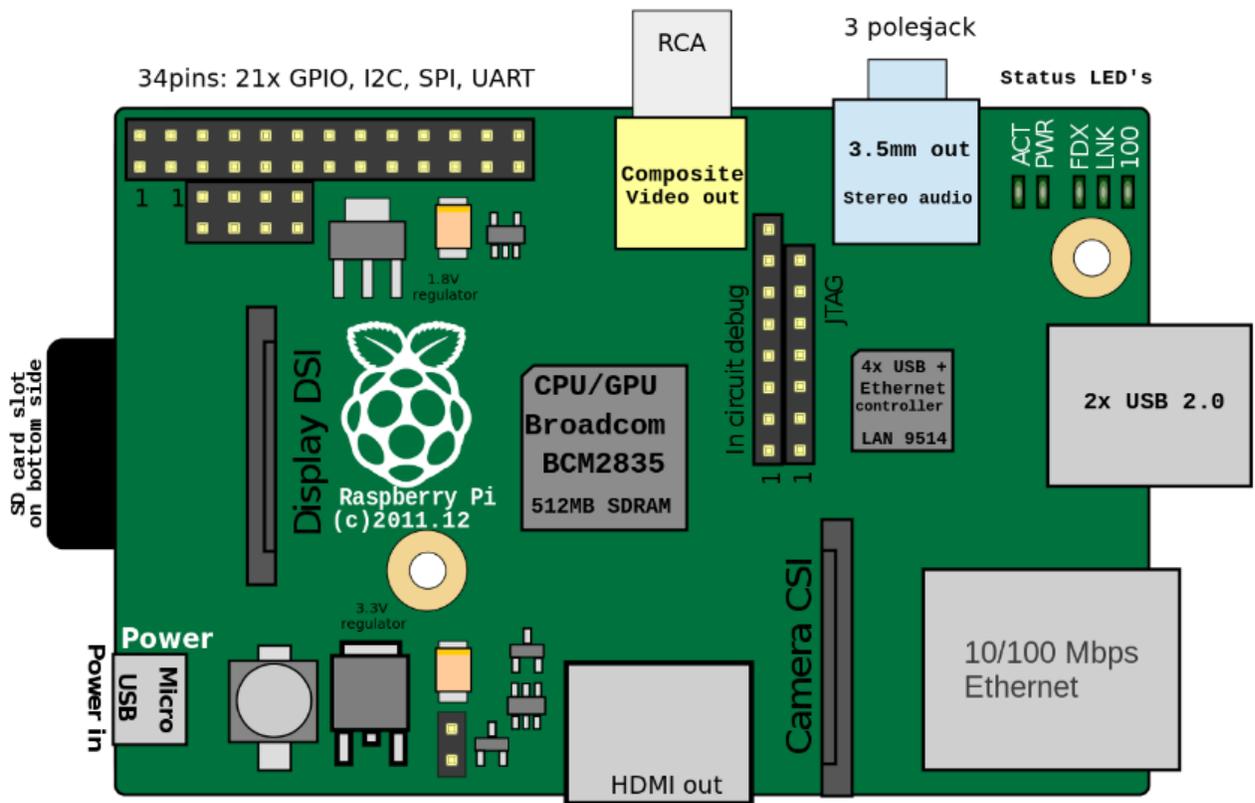


Ilustración 9: Diagrama que muestra los componentes de un microprocesador Raspberry Pi

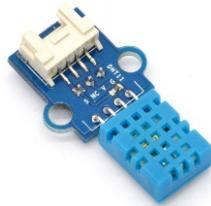
## Ejemplos de Sensores

Gracias al amplio apoyo que están ofreciendo las empresas desarrolladoras de sensores y actuadores a ambas plataformas, el catálogo de dispositivos compatibles es amplio y aumenta día a día. A continuación, enlistamos algunos de los sensores utilizados en el transcurso de este trabajo a modo de ejemplo e introducción al tema. Más información técnica puede encontrarse en las hojas de datos de dichos dispositivos, localizadas en la sección de anexos.

### Sensor DHT11

Este es un sensor digital que nos permite medir la humedad y temperatura ambiental, las que se obtienen en porcentaje y centígrados respectivamente.

Para conectarse cuenta con una entrada para cable de 4 pines en la parte frontal o con 3 pines en la parte posterior.



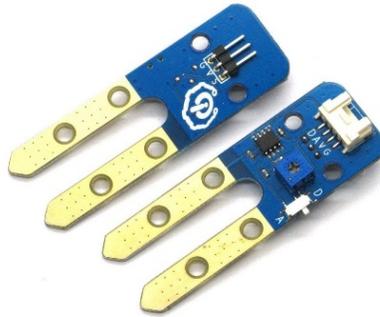
*Ilustración 10: Fotografía de un sensor de humedad y temperatura DHT11*

### Sensor Humedad de Superficie

Este sensor analógico/digital compuesto de dos probetas permiten medir la humedad de superficie, dando como resultado un número entre 0-1023 para la lectura analógica y entre 1 y 0 para la digital.

Si se utiliza de manera analógica, es necesario tener en cuenta que mientras más se acerque la lectura a 0, significa que está más húmedo y cuanto más cerca se esté del 1023, se encuentra más seco. Esto permite definir estados intermedios, de manera que se obtenga información más útil para nuestro propósito.

Para su conexión tiene disponible una entrada para cable de 4 pines o bien sus 3 pines principales en la parte posterior.



*Ilustración 11: Fotografía de un sensor de humedad de superficie.*

En este capítulo hemos podido conocer aspectos más técnicos de las plataformas de hardware libre que existen en el mercado y uno de los elementos que intervienen en la relación del Internet de las Cosas mostrada en el capítulo anterior.

Si bien es cierto que algunos de los aspectos que se han analizado a lo largo de este capítulo son temas de un área en la que no se ahonda a lo largo de la carrera de licenciatura en Informática, el conocer otros métodos de resolución de problemas permite al licenciado el poder generar soluciones enfocadas al área administrativa y/o contable con dichos elementos, es ahí donde radica la flexibilidad de la carrera. Así como en este capítulo revisamos herramientas de hardware, en el siguiente se abarcará el tipo de software que se utiliza por el área de especialización del licenciado, mostrando el panorama actual de dicho nicho y complementando algunas aplicaciones de las herramientas mostradas aquí.

## Capítulo 3

### SOFTWARE EMPRESARIAL

#### ¿Qué es el software?

En el capítulo pasado nos enfocamos a parte física de la tecnología, es decir, abordamos el hardware, pero para que éste se pueda utilizar es necesario el que exista un medio de comunicación que permita interactuar al usuario con el hardware: Este es el software.

Podríamos definirlo como la parte intangible de la tecnología que permite una interacción más precisa y sencilla con la misma, *“El software de computadora (...) abarca programas que se ejecutan en una computadora de cualquier tamaño y arquitectura, documentos que comprenden formularios virtuales e impresos, datos que combinan números y texto y también incluyen representaciones de información en audio y video”* (Pressman, 2002).

Como lo menciona Pressman consiste en los programas (Que son un conjunto de líneas de código que realizan una acción) que se ejecutarán, la documentación del mismo que permite informar sobre la funcionalidad y un diccionario de datos, el cual permite comprender lo que necesita el software como entrada de datos y cuál será su resultado.

Es necesario conocer un concepto un poco más general, los sistemas. Para los que Bertoglio propuso la definición: *“Entendamos como un sistema a un conjunto de elementos independientes que se encuentran relacionados entre sí y que interactúan con el fin de alcanzar un objetivo en común”* (Bertoglio, 1982). Definición que, se podría complementar con detalles según el contexto del sistema del que hablemos. Todas las escuelas coinciden en que son los requerimientos mínimos para clasificar a un conjunto de elementos como sistemas.

Esta noción de sistema es fácil de percibir en el software, los programas no son más que un conjunto de líneas de código que agrupadas permiten realizar una acción, estas líneas a su vez son sistemas más pequeños pues se componen de una serie de caracteres acomodados de cierta manera que obedece a un lenguaje particular.

Así mismo, el conjunto de programas o las acciones que estos realizan son únicamente una serie de pasos que permiten lograr una acción mayor: El programa encargado de leer las teclas numéricas

presionadas en un teclado, más el que permite la traducción de dichas teclas en números, más el programa que suma esos números y los muestra en una pantalla, dan como resultado que podamos sumar números y ser notificados del resultante de dichas operaciones.

Los sistemas tienen 5 características en particular:

- El ambiente o entorno: Se refiere al entorno en el que el sistema se desarrolla, todo aquello que está fuera del sistema, pero que interactúa con él.
- La frontera o límite: Son aquellos discriminantes que separan los elementos que le pertenecen al sistema de aquellos que pertenecen a otro.
- Las entradas: Se refiere a los recursos requeridos por el sistema para trabajar y poder lograr ese objetivo.
- Los procesos: Engloba todas las actividades y acciones que se deben realizar para cumplir o alcanzar dicho objetivo.
- Las salidas: Son todos los productos o aportaciones que hace el sistema al ambiente.

Debido a que en tecnología el principal objetivo de los sistemas es el procesamiento de datos para la generación de información que sea útil como soporte en la toma de decisiones de las organizaciones, a los sistemas en general se les conoce como sistemas de información. Dicho de otro modo "(...) Se encarga de entregar la información oportuna y precisa con la presentación y el formato adecuados, a la persona que la necesita dentro de la organización para tomar una decisión o realizar alguna operación y justo en el momento en que esta persona necesite disponer de dicha información" (Vieites, 2006).

Cabe destacar que en ningún momento se están mencionando elementos de hardware y software en dichas definiciones, esto es debido a que para la generación de un sistema de información no son necesarios dichos elementos: Para la generación de una nómina no se requiere de una computadora, sólo de un especialista que conozca el origen de datos (entrada) y el cómo estos son relacionados en dicho documento a través de las operaciones matemáticas y clasificaciones necesarias (proceso) para la generación del documento de nómina final (salida). Por ello es que se puede deducir que para tener un sistema de información es necesario contar con los procesos que en él intervienen, las personas que ejecutan dichos procesos y una base de datos que sirva de entrada a dichos procesos.

Cuando hablamos de sistemas y tecnología nos referimos a un tipo de sistemas de información muy particulares denominados como sistemas informáticos, los cuales incluyen el hardware y software a los procesos, las personas y los datos como elementos del mismo y que, al estar relacionados entre sí, buscan la generación de información como objetivo a través de su uso. Es en esta clasificación en donde cae todo el software desarrollado.

A pesar de que en la literatura especializada se refieran a ellos como sistemas de información, no hay que perder la noción de que en realidad es un subconjunto de esos, pues como mencionamos arriba, la definición de sistemas de información es utilizada por muchos ámbitos como puede ser el administrativo, geográfico, matemático, etc. Para fines prácticos, debido a que ya hemos expuesto sus diferencias nos apegaremos al común utilizado por dicha literatura que fundamenta esta investigación.

Cuando hablamos de software administrativo nos referimos al “Conjunto formal de proceso que, operando sobre una colección de datos estructurada de acuerdo a las necesidades de la empresa, recopila, elabora y distribuye selectivamente la información necesaria para la operación de dicha empresa y para las actividades de dirección y control correspondientes, apoyando, al menos en parte, los procesos de toma de decisiones necesarios para desempeñar funciones de negocio de la empresa de acuerdo a su estrategia” (Kenneth C. Laudon, 2012).

## **Breve historia de software empresarial**

Cuando hablamos de la historia del software empresarial, en realidad estamos hablando de la historia del software en general pues las necesidades de los negocios han sido el principal motor para el avance tecnológico: Muy a pesar de que durante las presentaciones de las primeras computadoras éstas ejecutaban programas de operaciones y cálculos matemáticos, sus primeras aplicaciones reales fueron para cálculo de nóminas, además de las aplicaciones científicas.

Los primeros softwares contables datan de los años 60's, al responder a la necesidad del área de contabilidad. Sin embargo, también responde a otro elemento: Debido a que dicha área se regula con base en leyes, normas y reglas, desarrollar este tipo de software era relativamente barato para la época pues el núcleo de software era el mismo para las empresas de una región particular. Podemos aseverar que con el nacimiento del software contable también nació una aproximación de

la creación en serie de software. Una vez que el software contable fue adoptado y obtuvo gran éxito, comenzaron a surgir las necesidades por parte del área administrativa: la gestión de pagos, facturas y cobros son algunas de las primeras aplicaciones que surgieron para dicho fin, incluso algunos de los cuales contenían integración con los sistemas del área contable, lo cual para la época representaba no sólo una enorme innovación, también una reducción de errores y costos.

La siguiente área que pudo beneficiarse con el software fue la gestión de inventarios con los primeros softwares de control de stock o ICS (*Inventory Control System*) con los que se podían conocer las existencias de cada producto en el almacén, el número de salidas y el momento o lapso en el que se dieron, así como su valoración o costo. De nuevo, la integración con otras aplicaciones de la empresa ofrecía mayores beneficios dentro de los procesos internos al optimizar los mismos.

Con la llegada de los 70's surgió un nuevo tipo de ICS, llamado planificador de los requerimientos de material (MRP por sus siglas en inglés, *Material Requirements Planning*) con la característica de permitir la elaboración del plan de materiales aplicando un enfoque jerárquico a la gestión de inventarios. Esto resultó en reducciones de inventario y tiempos de proceso y suministro para las empresas de la época. Su principal requerimiento era una precisión considerable en el plan de producción por realizar, ya que es la principal entrada de datos para dichos sistemas.

Entramos en la década de los 80's con la aparición de los MRP II o planificador de los requerimientos de manufactura (*Manufacturing Requirement Planning*), que es la evolución de los viejos MRP y que agregan funcionalidades que toman en cuenta también las necesidades de materiales para su fabricación. Sin embargo, estos sistemas eran exclusivos para las áreas de producción (Y más precisamente para industrias manufactureras) lo que limitaba el mercado potencial, aunque ofrecieran integración con otras aplicaciones existentes.

Fue hasta la década de los 90's que se volvió a dar una evolución de dichos sistemas, siendo reemplazados con el surgimiento de los sistemas de planificación de recursos empresariales o ERP (*Enterprise Resources Planning*) los cuales abarcaban la planificación de todas las áreas de una organización, y ofrecía una completa integración entre dichas áreas, además de abarcar una gama de industrias, por lo que un ERP podía ser utilizado sin problema por una fábrica, un centro de distribución o una empresa de servicios.

Una última evolución se produjo durante la década pasada con la creación del llamado ERP II o ERP extendido, cuya principal diferencia radica en los procesos que se contemplan. Dichos procesos son alterados debido a la implementación de modelos de negocio novedosos utilizando tecnología para ello: Los modelos generados del llamado e-business (business to business, business to customer, etc.) dejan de ver a la empresa como una organización tradicional, pues su valor se enfoca más a la relación que tiene con los proveedores y clientes en su cadena de valor, por lo que han surgido una serie de nuevos requerimientos que no pueden ser abordados por los ERP tradicionales.

## **Clasificación del software empresarial**

Hablar sobre una clasificación de los sistemas empresariales es un tema un tanto confuso. Oficialmente no existe una clasificación. Sin embargo, debido a que estos sistemas van dirigidos a un área de la organización en particular, se ha tomado a las propias áreas como una clasificación y ha sido ampliamente aceptada en la industria. Podemos encontrar sistemas muy especializados para control de inventarios, facturas, compras, ventas, etc. que están enfocados a organizaciones de medio a gran tamaño.

Por otro lado, se puede encontrar software integrador como lo pueden ser los ERP's (Enterprise Resource Planning) enfocados a las micros y pequeñas empresas que necesitan un control, pero no requieren tanta complejidad para la realización de sus operaciones. Es por ello que en este tipo de software contamos con módulos generales que permiten cierta personalización para acoplarlo a dicha organización.

Existen ciertas clasificaciones de dichos sistemas basadas en la función que estos tienen dentro de la organización, no confundir con la funcionalidad que estos incluyen. La clasificación propuesta por Kenneth y Jane Laudon en su libro "Sistemas de Información Gerencial" es un ejemplo de estas, en donde las categorías se sustentan en el valor que dichos softwares aportan a los procesos del negocio, más que la funcionalidad de estos, dicha clasificación es la siguiente:

- Sistemas de procesamiento de transacciones (Transactions Process Systems o TPS): Se encargan de la gestión de información referente a las transacciones producidas en el ejercicio diario de la organización.

- Sistemas de Información Gerencial (Management Information Systems o MIS): Enfocados en proporcionar un “diagnóstico general” del estado actual de la organización.
- Sistemas de Soporte a Decisiones (Decision Support Systems o DSS): Enfocados al análisis y proyección de variables y escenarios de negocio con el fin de evaluar y proporcionar apoyo a la toma de decisiones.
- Sistemas de Información Ejecutiva (Executive Information Systems o EIS): Enfocados a brindar información del estado de una unidad, sucursal o área en particular a los recursos gerenciales de la organización.
- Sistema de Automatización de Oficinas (Office Automation Systems u OAS): Enfocados al apoyo a las actividades diarias del personal administrativo de la organización.
- Sistemas Expertos (SE): Enfocados a emular el comportamiento de expertos en un tema particular.
- Sistemas Planificadores de Recursos Empresariales (ERP): Enfocados a la integración de los procesos e información generada por los mismos de toda la organización en un mismo sistema.

A lo largo de este capítulo hemos dado un paseo por la historia del software y una de las áreas de especialización importantes con las que cuenta. Hay un detalle importante a resaltar, y es que, aunque existe software enfocado a diversas áreas organizacionales, éstos caen en la clasificación propuesta por los autores Laudon, por lo que ambas visiones pueden utilizarse en conjunto, complementando el porqué de la necesidad de software especial para las empresas.

Con lo revisado en los capítulos precedentes se ha construido un modelo que permite la conjugación de diferentes tecnologías y aspectos para el desarrollo de soluciones con enfoque de Internet de las Cosas y que será el objetivo del capítulo siguiente, pero más importante aún es lo que este modelo representa: Un puente que comunica los avances tecnológicos/ingenieriles con el mundo de los negocios y las empresas, dos mundos que generalmente son tratados por separados, pero que la historia demuestra la dependencia entre ambos existente.

## Capítulo 4

### MODELO DE REFERENCIA

A lo largo de este capítulo se presentará un modelo de referencia a seguir para la implementación de una solución de IoT, los elementos que lo conforman y la interacción que tienen entre sí. Con el fin de ofrecer una visión sobre el potencial de este modelo y brindar un panorama del futuro que tiene en el sector empresarial, se presentan los beneficios que puede aportar la aplicación del modelo a aspectos del área de TI y de la vida cotidiana,

#### Motivos y origen del modelo

Vale la pena destacar dos situaciones que ocurren constantemente en el desarrollo de propuestas basadas en un enfoque IoT:

1. La primera es la delimitación en áreas de aplicación sobre las que se están trabajando. Esto se refiere a que en la actualidad las personas que trabajan con plataformas como Arduino o Rasperry Pi generan soluciones que pueden clasificarse como de ingeniería. Arreglos de Leds que encienden y apagan, medidores de presión, entre otros; por otro lado existen otro grupo de personas (Llamados a sí mismas “*Makers*”, proveniente de un movimiento cultural homónimo cuya ideología se basa en una necesidad del ser humano para *crear* herramientas que permitan mejorar su vida, así como el compartir dicho conocimiento con sus semejantes) que se dedican a la creación de prototipos para uso propio: aquí encontramos una gran cantidad de “herramientas” como lo pueden ser la automatización parcial o total de lugares como invernaderos, cuartos de mantenimiento o de acciones a bajo costo como puede ser la apertura o cierre de puertas de habitaciones, oficinas o garajes.
2. La segunda situación consiste en la falta de estándares para la aplicación del paradigma de IoT debido a su auge reciente, aunque en los últimos meses han surgido iniciativas y propuestas de estándares, éstas llevan su tiempo de revisión. Una medida que han utilizado en este medio ha sido el uso de algunas generalizaciones que la propia comunidad ha establecido con el fin de hacer un poco más sencilla la curva de aprendizaje de quienes se

interesan en aprender a utilizar las plataformas, así como para el soporte para quienes realizan proyectos más ambiciosos.

Estas dos situaciones son las principales motivaciones para el desarrollo de este trabajo, que tiene como propósito la propuesta de un modelo que permita desarrollar soluciones basadas en IoT y que proponga una solución para las situaciones antes mencionadas: Con este modelo se pretende romper con la exclusividad de dichas aplicaciones para el ámbito de la ingeniería. Pues si algo nos ha enseñado la historia (Y más la historia de la computación) es que la ingeniería y los negocios no están peleados sino todo lo contrario, son dos vertientes que se complementan de manera casi romántica.

Con respecto a la segunda situación, fue necesaria la consulta del estado del arte (Si se permite dicha expresión debido a la relativa novedad del IoT) con el fin de identificar qué elementos se utilizan y cuáles son comunes entre proyectos y prototipos existentes, pues a partir de estas relaciones se podría generar un “esqueleto” del cómo se construyen y operan actualmente dichos proyectos.

El resultado de esta tarea arrojó 3 puntos principales:

- 1) Todos los proyectos consultados se basaban en la necesidad de interactuar dentro del ambiente donde se encontrarán, ya sea de manera pasiva (con sensores) o activa (con actuadores).
- 2) Se necesitaba un medio de comunicación que permitiera el envío y recepción de datos entre dos puntos geográficos, donde se encontraban los entes que le interesaban medir a un sujeto y donde se encontrara dicho sujeto, el cual generalmente se trataba de Internet.
- 3) El sujeto interesado necesitaba la presentación de la información contenida en dichos datos con el fin de poder tomar decisiones, para ello hacía uso de servicios de almacenamiento y presentación en la nube que él mismo programaba, o simplemente (este engloba los menos casos) revisaba los datos “crudos” obtenidos desde un portal web o una aplicación para su teléfono inteligente.

Contando con la información expuesta anteriormente se diseñó un modelo que puede servir de base para la arquitectura de soluciones aplicando el IoT. El objetivo es aportar un esquema que proponga un orden que consiste en una división por capas independientes con diferentes propósitos y

responsabilidades definidas que, aunque funcionan como un todo, reducen la complejidad generada por la alta cantidad de interacciones entre diferentes tecnologías y protocolos a utilizar.

Es también importante mencionar que este es un modelo general, si bien es cierto que hemos dirigido este trabajo de investigación a las organizaciones lucrativas (a empresas, vamos) esto no quiere decir que no pueda aplicarse a cualquier organización que tenga procesos y un modelo de negocio que no necesariamente priorice la generación de capital.

Por última observación es que ni este modelo, ni el paradigma en general representan la nueva panacea tecnológica, sino todo lo contrario, tal como las actuales metodologías y arquitecturas de sistemas de información, es necesaria una etapa de levantamiento y análisis de requerimientos para concluir no solamente si es posible generar una solución de IoT, también el concluir si dicha solución aportara un valor significativo a la organización.

## Un modelo de referencia basado en capas

El modelo consta de nueve capas, que van desde los conceptos más tangibles (tan tangibles como lo pueden ser los objetos físicos) hasta abstracciones más complicadas (Porque el conocer y entender no solo los procesos de una organización, si no el cómo estos interactúan y apoyan el modelo del negocio no es tarea sencilla) en una estructura que, visualmente, parece una pila similar a la del modelo OSI.

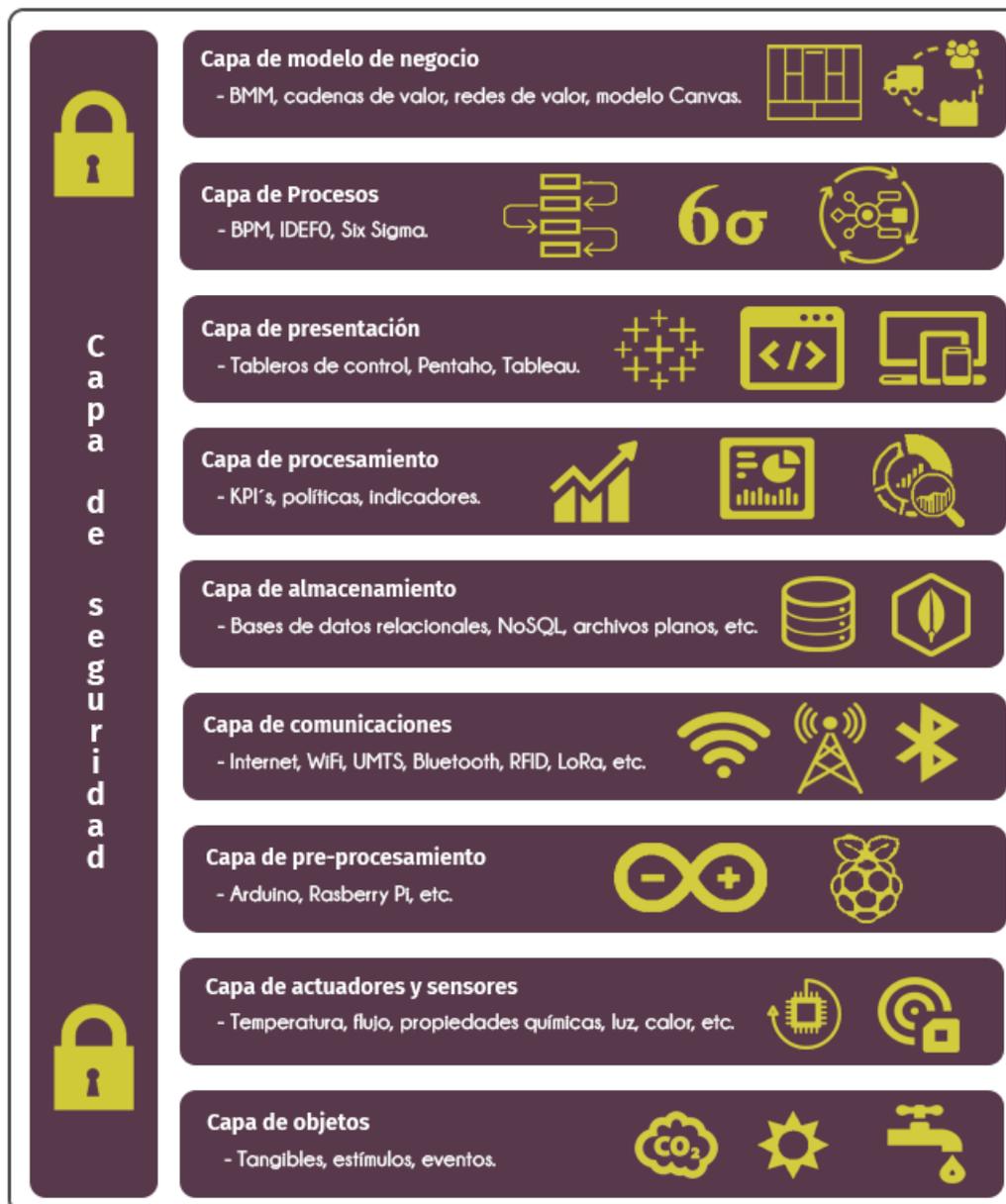


Ilustración 12: Diagrama del modelo de referencia propuesto

Las capas que integran el modelo son las siguientes:

1. Capa de Objetos: Para generar una solución útil a una organización y en especial en el desarrollo de sus procesos de negocio es necesario conocer qué variables y eventos del ambiente son importantes para el proceso, es por eso que esta capa engloba todos los objetos (tangibles e intangibles) que son relevantes. Aquí cabe hacer un señalamiento, cuando se habla de objetos no se refiere ni a las cosas que estarán en constante comunicación, ni a objetos de la vida cotidiana como sillas, habitaciones u hogares; se refiere a las fuentes que proveerán de datos relacionados a sus atributos, como pueden ser la temperatura, humedad, presión, sonidos, eventos generados, etc.

2. Capa de Actuadores y Sensores: Una vez identificados los objetos de interés y los atributos relevantes de éstos, es necesario identificar el tipo de dispositivos que permitan medirlos, es decir, si dichos atributos pueden ser monitoreados con un tipo de sensor y cuál es la precisión necesaria.

Es importante en esta capa identificar si el proceso requiere interactuar con otros objetos, actuar ante ciertos eventos o estímulos, qué tipo de reacción es necesaria y cuáles dispositivos son requeridos para dicha tarea: motores, servomotores, relevadores, fuentes de poder externo, etc.

3. Capa de Pre-procesamiento: Así como sucede en las computadoras personales y otros dispositivos similares que para su funcionamiento correcto necesitan un “orquestador”; en este caso es la tarjeta madre. Para el modelo IoT existen orquestadores de dos tipos: placas microcontroladores y placas microprocesadores. En términos generales, ambas ofrecen las interfaces necesarias para suministrar energía, directivas y realizar la gestión de memoria para el funcionamiento correcto de los ambientes generados. Su diferencia radica en la capacidad de procesamiento que tienen a su disposición. Los microcontroladores poseen recursos limitados, pero permiten una rápida respuesta y mejor manejo en memoria, al trabajar con datos a bajo nivel. Los microprocesadores proveen una capacidad de procesamiento superior, sacrificando la simplicidad y el manejo de memoria eficiente.

4. Capa de Comunicaciones: Este es uno de los componentes que permite conceptualizar el IoT, que actualmente no se limita exclusivamente a Internet, sino que se traslada a otras tecnologías, también de bajo costo, que permiten lograr un ambiente siempre conectado. Si bien es cierto que se trabaja con protocolos de intercambio de información como http, ftp o el recientemente aceptado como estándar MTQQ (Estándar de bajo consumo desarrollado por IBM) además de ser compatible con formatos como Json o XML, también es cierto que puede trabajar a través de Bluetooth, de redes GSM e inalámbricas, incluso a través de RFID que permite una mayor compatibilidad entre dispositivos, lo que se traduce en un mayor número de ambientes sobre los cuales trabajar.
5. Capa de Almacenamiento: Esta capa engloba las formas en las cuales se almacenan para su procesamiento posterior los datos transmitidos. Aquí se pueden encontrar estructuras como bases de datos estructuradas, pero también modelos no estructurados como No SQL u otro tipo de contenedores como archivos planos, CSVs, hojas de cálculo, utilizar servicios de almacenaje en la nube o incluso el uso de servicios o plataformas como Google App Engine, que permiten mandar directamente los datos a alguna de las aplicaciones que forman el ecosistema de Google y ahí lograr la persistencia de los mismos.
6. Capa de Procesamiento: Esta capa se ocupa del procesamiento de los datos recibidos con el fin de traducirlos, agruparlos e interpretarlos para la generación de información útil al usuario final. En esta capa se engloban las rutinas necesarias para el mantenimiento de los propios datos: Generación de medidas estadísticas como promedios, máximos o mínimos, procesamiento por bloques que representen intervalos de tiempo como horas, días, semanas o meses, la generación de indicadores que permitan una evaluación de los datos obtenidos y con ello una evaluación del ambiente, ejemplo de estos son los KPI's (*Key Performance Indicator*, en español Indicador clave de rendimiento) los cuales permiten medir el rendimiento de una tarea que se encuentra en ejecución con el fin de auxiliar en la toma de decisiones.

7. Capa de Presentación: Esta capa contiene los diferentes medios que permiten la publicación de la información generada en la capa anterior: Los reportes, gráficas, estadísticas y comparativas se presentan aquí, generalmente estos son accedidos a través de algún sitio en internet como puede ser un portal o una plataforma en particular. Así mismo se hace uso de herramientas de inteligencia de negocio que ofrecen más características que la simple generación visual de información. Permiten la generación de tableros de control dinámicos, así como estructuras en las cuales el usuario final puede manipular según su necesidad inmediata como lo son los cubos de información o las tablas dinámicas, herramientas que pueden ser la suite Pentaho, Tableau, etc.
  
8. Capa de Procesos: A pesar de que la capa de presentación puede llegar a generar, presentar y publicar información útil de múltiples maneras, el esfuerzo será en vano si esta información no aporta nada útil para los involucrados en los procesos donde interviene la solución basada en este modelo, esa es la función de esta capa, reafirmar que la solución es un apoyo que viene a soportar un proceso en particular. Este apoyo puede darse de diferentes maneras: Como parte de las actividades que forman el proceso por ejemplo el cuantificar de manera exacta el volumen de líquido extraído de un contenedor, o puede ser como parte de la validación del resultado de un producto e incluso como parte de los procesos de calidad del mismo, al apoyar en la obtención de estándares de la industria tales como Six Sigma, alguna norma ISO aplicable a la empresa, entre otros, esto sin perder de vista que la información generada por la solución está enfocada a la toma de decisiones en la búsqueda del cumplimiento de metas y objetivos que obedecen a planteamientos contenidos en diseños de procesos basados en disciplinas como la gestión de procesos de negocio y modelos de tomas de decisión como puede ser IDEF0 entre otros.
  
9. Capa de Modelo de Negocio: Si bien es cierto que en la capa de procesos es donde comenzamos a dar un enfoque administrativo a nuestra solución, los beneficios de esta mejora de procesos son reflejados directamente en la dirección del negocio, por lo que toda la infraestructura está pensada para aportar valor y con ello una mejora significativa a los

objetivos y metas fijadas en el modelo de negocio y que estén alineadas a la visión del mismo. Esta capa representa la maduración no solo de la tecnología, si no del modelo propuesto y el modelo de negocio en sí, pues si este último no está bien definido, se desconoce o es inexistente, nuestra solución puede ser útil a los procesos de negocio y desempeñarse adecuadamente y aun así no cumplir con el objetivo con el que fue creada. Herramientas como el modelado Canvas, el modelo de incentivación del negocio o el diseño de cadenas o redes de valor son técnicas de relativa sencillez que permiten una visión general del negocio al que se aplican, facilitando a su vez el diseño y aplicación con el modelo aquí propuesto.

Como puede notarse el modelo pretende abarcar todos los aspectos que notamos en los proyectos existentes, complementar o enriquecerlos, agregar otros e incluso es configurable. Es decir, según la situación que se intente resolver puede ampliarse o reducirse, de tal modo que lo vuelve flexible al cambio y compatible con metodologías de desarrollo de sistemas y administración de proyectos como lo puede ser SCRUM, Open Up, RUP, etc.

## **Seguridad**

La seguridad es uno de los principales temas y preocupaciones durante el desarrollo de sistemas, ya que según las características del producto varía la estrategia sobre cómo abordarlo. En respuesta a esta diversidad han surgido diferentes áreas especializadas en los componentes que integran un sistema: Seguridad en bases de datos, telecomunicaciones, para accesos en los sistemas operativos, los protocolos de comunicación, etc.

A diferencia de los modelos de desarrollo de software convencionales, el Modelo que se presentó en la sección anterior enfrenta un reto mayor en cuanto a seguridad se trata, esto es debido a que las múltiples capas que lo componen, las cuales no hacen referencia solamente a componentes de software, sino también a componentes de hardware e incluso de negocio que interactúan de manera continua.

Es por eso que el tema propiamente es sumamente extenso, sin embargo, puede para efectos prácticos a la seguridad que la industria de TI ha ido desarrollando, lo que significa que se puedan aplicar estándares o técnicas de seguridad definidas por algún área en particular.

A continuación, podemos mencionar algunas de las necesidades de seguridad requeridas por el modelo, en la capa de sensores la seguridad viene en forma de los datos que obtenemos de ellos según la precisión y el formato en el cual la proveen pues, aunque el sensor ofrezca una gran precisión, si este requiere de alguna transformación es imposible el no sufrir una pérdida. También en esta capa podemos incluir la seguridad en la infraestructura con que se relaciona: Si las conexiones que se tiene están mal hechas o son de mala calidad, puede comprometerse no solo el sensor, también otros sensores o incluso el origen de datos que toma.

La capa de pre-procesamiento es similar a la capa de sensores debido a que la seguridad también es requerida en dos aspectos: El de los datos que obtiene, procesa y transforma, por lo que el acceso a los dispositivos de esta capa debe estar controlado a través de algún mecanismo como la autenticación (Como es el caso de la Arduino Yún, que para acceder a sus configuraciones y a su núcleo Linux es necesario la configuración y autenticación con un usuario y contraseña); Por otro lado la seguridad en la infraestructura que se tiene relacionada es también de vital importancia.

La capa de telecomunicaciones también tiene influencia en la seguridad de la información que por ella se transmite, aquí entran protocolos de cifrado para el contenido de los paquetes que se envían. Por otra parte, se encuentra la seguridad del medio por el que se envía, el cual incluye protocolo de construcción de cables, arquitecturas de red y diferentes protocolos de envío y recepción como lo son el X20 o TCP/IP.

En la capa de persistencia se enfoca a la seguridad de información. En ella se tienen mecanismos de acceso a las bases que almacenan los datos recibidos, así como a las conexiones que se establecen con esta e incluso las consultas que se realizan, a modo de controlar el acceso a información que puede ser sensible. A esta capa se le añaden la capa de procesamiento y presentación pues están relacionadas en el aspecto del acceso a los datos, es necesario asegura que solamente las aplicaciones permitidas puedan consultar lo almacenado y puedan mostrarlo para los usuarios. Estas tres capas componen los recursos importantes que la seguridad del servidor donde residen se

encarga de cuidar: Seguridad en los accesos a dicho servidor, en la red de comunicación para evitar accesos no permitidos, en la administración de mantenimiento y respaldos, etc.

Finalmente están las dos capas enfocadas al negocio las cuales no están atadas a ningún área o concepto de la seguridad informática, más bien a herramientas financieras más afines a la viabilidad o rentabilidad de la solución según el impacto que tiene en el proceso de negocio. Para fines de este trabajo delimitaremos estas capas a la utilidad que tenga la solución dentro del proceso y el impacto de este cambio dentro del modelo de negocio.

## **Beneficios**

Los beneficios del uso de este modelo de referencia en el desarrollo de proyectos basados en IoT son varios, principalmente es el dotar de un orden al desarrollo, repercutiendo en su complejidad inherente, lo que siempre es un enorme beneficio.

Por otro lado, se encuentra cierta seguridad sobre la utilidad de dicho desarrollo: si éste se plantea y construye tomando en cuenta las últimas dos capas, es decir, alineado al negocio con el fin de aportarle un bien, será bien recibido en la organización (lo mismo que cualquier sistema que se pretenda implementar).

Finalmente, el mayor beneficio es para el desarrollador debido a que, al generar soluciones basados en otras ideologías no solo amplía su visión, también lo hace su forma de pensar y de abordar problemas, pues más que un conjunto de objetos interconectados, el paradigma IoT consiste en una nueva forma de resolver problemas reales, actuales, que los sistemas no pueden o les es complicado el abordar.

## **Iniciativas Extranjeras**

Actualmente en el mundo, tanto Arduino como Raspberry Pi, se utiliza con diferentes propósitos: Desde su acercamiento a niños de primaria y secundaria con el fin de facilitar en ellos la enseñanza de la programación y los lenguajes utilizados, hasta algunas personas de todas las edades interesadas (los llamados “makers”) quienes han creado espacios dedicados a instruirse y generar sus propios

prototipos e intercambiar experiencias. Incluso comparten con sus colegas las instalaciones (makerspaces).

En Estados Unidos, por ejemplo, año con año la revista Make realiza ferias donde las personas presentan sus creaciones al público asistente e incluso hay competencias que buscan el apoyar y mejorar dichos proyectos. Existen iniciativas por parte del gobierno inglés para dotar a cada niño de una Raspberry Pi y de incluir materias de programación en sus planes de estudio, con el fin de dotar de habilidades a los pequeños que mejoren su perfil educativo, iniciativas muy similares a los talleres de robótica y Arduino con el uso de Lego, con el fin de hacer más atractivo para los niños el acceso a estos conocimientos.

## **Iniciativas Nacionales**

Hablando de lo que se hace dentro de nuestro país poco a poco éstas tecnologías llaman la atención del gobierno y particulares por igual. Por un lado, se han formado en los últimos años unos cuantos makerspaces en la ciudad de México y algunos otros estados, como lo pueden ser “Hacedores México” o “330 Ohms Makerspace” que pretenden difundir el uso de estas tecnologías entre la población. El surgimiento de iniciativas por parte del Instituto Nacional del Emprendedor (INADEM) que apoyan a dichos makerspaces a través de convocatorias para la generación de prototipos y soluciones en dichos espacios.

La educación tampoco se queda atrás, pues instituciones como la Facultad de Ingeniería de la UNAM o el Instituto Tecnológico Autónomo de México (ITAM) han comenzado a incluir estas plataformas en sus planes de estudio, de manera que ofrecen a los universitarios nuevos conocimientos y nuevas herramientas que complementan su formación académica. Desafortunadamente se encuentra limitado a las ingenierías relacionadas con las ciencias de la computación, motivación origen del modelo de referencia presentado en este capítulo.

A lo largo de este capítulo se ha presentado un modelo de referencia que permite la construcción de soluciones que implementan Internet de las Cosas y son enfocadas a ayudar a las organizaciones en sus procesos de trabajo, a manera de un plano basado en componentes que no están atados a alguna tecnología o proveedor. En particular, retomando la metáfora del puente, una comunicación que se basa en 9 etapas con las cuales se pretende poder traducir de herramientas de bajo costo

muy enfocadas a ingeniería, a herramientas muy administrativas que sean significativas para el negocio y útiles en la toma de decisiones. En este trabajo se llevó el concepto del modelo de referencia a probar, como se verá más adelante, a través de la generación de dos prototipos que responden a problemáticas reales en micro y pequeñas empresas.

## Capítulo 5

### PRUEBAS DE CONCEPTO

Mediante dos pruebas de concepto se pretende demostrar la viabilidad, utilidad y aplicación real del modelo de referencia presentado en el capítulo anterior.

Para fines de este trabajo de investigación aplicada, entenderemos por prueba de concepto la demostración de una idea o concepto a través de un prototipo funcional, con el fin de exponer su utilidad real y aplicaciones posibles.

Ambas pruebas se realizaron con el apoyo de la metodología Open UP, que consiste en una metodología ágil que implementa algunas de las características de RUP (Rational Unified Process) y que se basa en el concepto de micro incrementos, avances personales y rápidos que permiten progresar en conjunto a través de las diferentes fases del ciclo de vida del proyecto.

Se eligió la metodología Open UP por su flexibilidad, ya que al ser configurable permite tomar los elementos más adecuados al caso a tratar, haciéndola compatible con una cantidad diversa de tipos de proyectos. Además de ofrecer un enfoque en donde el valor del producto final tiene un gran peso en el desarrollo del mismo, este enfoque empata con las últimas dos capas del modelo de referencia propuesto.

Es oportuno mencionar que toda la documentación derivada del desarrollo de las pruebas de concepto presentadas se encuentra disponible en los anexos de este trabajo de investigación, acompañado de los diagramas correspondientes al diseño de la solución. Son estos anexos la prueba de que el modelo de referencia aquí presentado puede trabajar junto a una metodología de desarrollo de sistemas y crear productos con un nuevo enfoque pero que tienen un respaldo de buenas prácticas y la experiencia de la industria que llevó a la generación de dichas metodologías de trabajo.

## **PC1: Mejora de mediciones ambientales en un invernadero**

### **Capa de negocio y proceso**

Se requiere la generación de un prototipo que integre hardware y software con un enfoque de Internet de las Cosas, que permita conocer el estado del ambiente en un invernadero a través del censado de ciertas variables y su posterior consulta a través de un portal web.

Con esto se pretende apoyar el proceso de operación y mantenimiento del invernadero, al proporcionar información exacta, precisa, correcta, a tiempo y costo competitivo para las personas involucradas en estas actividades, con el fin de apoyar a la toma de decisiones.

Las actividades de mantenimiento del invernadero se realizan con técnicas empíricas donde la experiencia de la propietaria es lo dominante.

A pesar de tener algunos años en funcionamiento y volverse un lugar completamente autosustentable, las características geográficas y climatológicas de la zona, aunado a la forma de operación puede representar una pérdida en la calidad de los cultivos.

Más información puede encontrarse en el *documento de visión* de esta prueba de concepto, localizada en la sección de anexos.

### **Capa de Objetos**

Tras pláticas con la dueña del invernadero sobre los problemas a los que enfrentaba durante el ciclo de siembra y cosecha del lugar, se delimitaron los objetos de interés a cuatro variables:

- Humedad ambiental
- Temperatura ambiental
- Humedad de la tierra
- PH de la tierra

Esto debido a que la ubicación del lugar se encuentra a una altura considerable sobre el nivel del mar, por lo que factores como la presión barométrica y la zona geográfica afecten de manera directa a estas mediciones, y con esto el desarrollo de los productos ahí producidos.

## Capa de Sensores

Para la construcción de esta prueba de concepto se utilizaron sensores del tipo “Electronic Bricks”, que son sensores montados sobre placas que permiten una interconexión sencilla mediante herramientas como protoboards y lectoras, e incluir resistencias para protegerlos ante un cambio de voltaje, simplificando su interconexión.

Se utilizaron en total 8 sensores de tres tipos:

1. Tres sensores digitales modelo DHT 11, encargados de monitorear el porcentaje de humedad ambiental (con una precisión de +-4%) y temperatura ambiental (con una precisión de +- 3°C).
2. Cuatro sensores analógicos modelo IM121017001, encargados de la lectura de humedad de superficie, a través de dos diodos que reaccionan al entrar en contacto con la tierra. La intensidad de esta reacción se ve reflejada en la señal producida.
3. Una probeta medidora de PH para suelos de BlueLab, este funciona a base de dos calibraciones, que dan los rangos de medición, de forma que estos pueden cambiar según los líquidos utilizados. Para poder utilizar la probeta es necesario contar con un lector BNF que permita la conexión con el microcontrolador.

Para distribuir los sensores a lo largo del terreno se decidió que solo los sensores DHT quedaran fijos y el resto se pudieran mover, por lo que se utilizó cable compuesto para minimizar la intervención del ambiente en la comunicación con el microcontrolador, esta distribución se muestra a continuación. En caso de necesitar más información respecto a dichos dispositivos, se recomienda el consultar los datasheets correspondientes<sup>6</sup>.

---

<sup>6</sup> Debido a que los datasheets de los dispositivos similares pueden variar según el vendedor, aquí se hace referencia a las utilizadas durante el desarrollo de las pruebas de concepto: DHT 11 <https://goo.gl/T9Utmy>, IM121017001 <https://goo.gl/P2Ewwd>, Probeta BlueLab <https://goo.gl/WjVHKl> y lector BNF <https://goo.gl/6h8Xy9>.

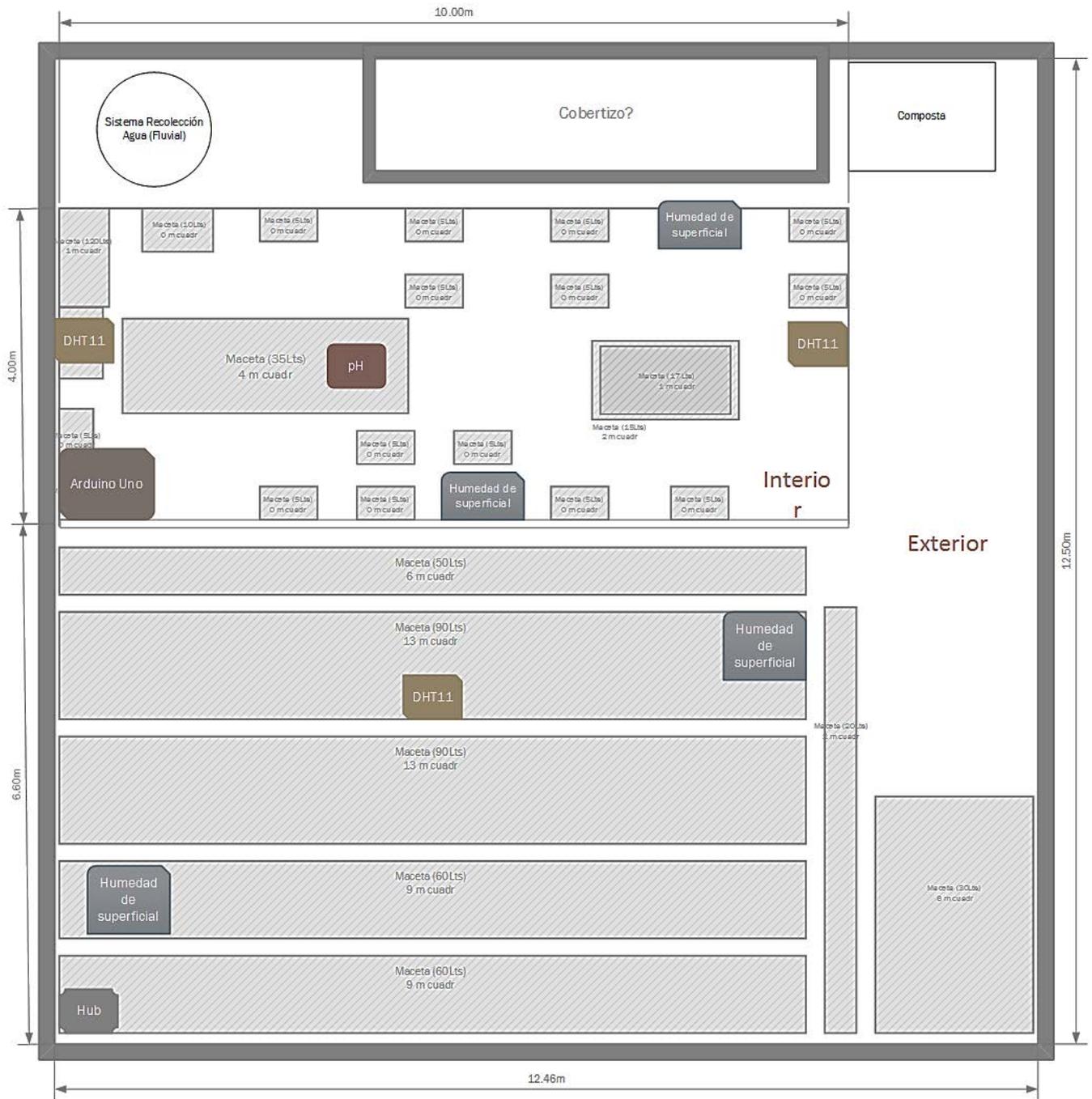


Ilustración 13: Distribución de los sensores sobre el terreno.

## Capa de Pre-Procesamiento

Para poder coordinar la obtención y el envío de lecturas de los sensores al resto de la arquitectura, el microcontrolador utilizado fue un Arduino Uno, cuya programación se divide en dos bloques.

Primero encontramos un bloque de configuración, que se encarga de revisar, adaptar el ambiente en el que se encuentra el microcontrolador y de asignar valores a variables y configuraciones internas.

1. Se configura el modulo para tener acceso a Internet desde el dispositivo.
2. Se asignan las direcciones IP, puerto y (si aplica) información adicional de dominio sobre el servidor destino a donde se estarán enviando las lecturas obtenidas.
3. Se hace la asignación de los pines que leerán los sensores, así como los modos o actividades que realizarán estos.
4. Finalmente se configura la “planificación” de la toma de lecturas, esto es, cada cuanto tiempo se realizarán dichas tomas.

El segundo bloque consiste en las lecturas de sensores.

5. Internamente la placa toma el tiempo que ella lleva encendida, con este dato se realiza la simulación de un “reloj interno”, lo que permite ejecutar tareas periódicas, como es el caso de la toma de lecturas.
6. Debido al número de sensores, su organización está dada para que cada n minutos se tome la lectura de todos los sensores de un mismo tipo, a manera de bloque: Los de temperatura y humedad cada 10 min, humedad de superficie cada 5 min y PH de suelo cada 30 min.
7. Cuando llega el momento de ejecutarse, la placa va a los pines específicos y obtiene dicha lectura en tiempo real, que se almacena temporalmente mientras se realiza la misma operación con el resto de los sensores de su tipo.
8. Una vez obtenidos, las lecturas son interpretadas por el microcontrolador según su tipo con el fin de traducir las señales obtenidas. Dicha interpretación puede darse a través de librerías o de manera manual a través de fórmulas matemáticas.

9. Finalmente, los resultados “interpretados” se almacenan en conjunto para poderse enviar.

El detalle de este flujo puede consultarse en las especificaciones de caso de uso, dentro de la sección de anexos.

## **Capa de Comunicaciones**

Una vez que se tiene un conjunto de lecturas interpretadas, estas se preparan para el envío hacia el servidor destino donde se encuentra la base de datos que almacenara las lecturas. Este proceso se realiza de la siguiente manera:

Los datos interpretados son convertidos a un formato JSON para reducir su tamaño total, simplificar su envío y manipulación a nivel servidor. Una vez generado este objeto JSON, se procede a crear una conexión con el servidor destino.

Si esta conexión es exitosa, se procede a añadir a un “paquete vacío” las cabeceras necesarias para que pueda viajar y llegar a través del estándar HTTP a nuestro servidor. Entre la información usada se puede destacar el uso del protocolo TCP a través del método POST, donde se indica que en el cuerpo se está enviando un objeto JSON a nuestro servidor, se estipula la ruta a la cual lo estamos enviando y cuyo remitente es un dispositivo de nombre “Arduino”.

El detalle de este caso de uso puede ser consultado en la especificación del caso de uso “Enviar lecturas” y en el diagrama de despliegue dentro de la sección de anexos.

## **Capa de Almacenamiento**

Del lado del servidor destino ocurre un flujo de datos que consulta un catálogo para complementar la información enviada, antes de guardarla:

1. Se obtiene el contenido enviado del almacenamiento temporal, esto es una cadena de texto que representa los datos enviados en formato JSON.
2. La cadena enviada es transformada en un objeto para simplificar su consulta.
3. Por cada uno de los objetos contenidos en el arreglo localizado dentro de la estructura enviada, se obtiene el número del pin del cual se obtuvieron dichas lecturas.

4. Con ese número de pin, se consulta el catálogo de sensores para obtener el identificador único del sensor asociado a ese pin.
5. Se preparan las consultas de inserción a la base de datos con los datos enviados y los identificadores obtenidos.
6. Las consultas son ejecutadas y los valores guardados en la base de datos.

Para más detalle puede consultarse el diagrama entidad relación localizado en la sección de anexos.

## **Capa de Procesamiento**

El procesamiento de las lecturas recibidas y almacenadas previamente se divide en dos tipos:

- ~ El primero es un tratamiento estadístico, en el cual se generan promedios, valores máximos, mínimos y desviaciones estándar de las lecturas recibidas a través de vistas que dividen al conjunto por momento del día: Mañana, Tarde y Noche. Así mismo, durante las 00:10 Hrs del día se ejecuta un procedimiento almacenado que obtiene todas las lecturas del día anterior y obtiene las mismas medidas estadísticas de todas las lecturas obtenidas el día anterior, por tipo de lectura, este último como forma de agilizar la obtención de reportes posteriormente.
- ~ El segundo es el realizado tras una conversación con la dueña del lugar: Ella mencionaba que le interesaría conocer el estrés por el que eran sometidos sus cultivos durante el día. Tras una investigación documental se concluyó que el mencionado estrés se trataba de una relación entre la temperatura y humedad ambiental, las cuales tenían influencia en la velocidad con la que un cultivo pierde agua.

Este concepto se tradujo a un indicador, con valores del 1 al 8 y que se lee como “mientras más alto es, el cultivo está siendo sometido a más estrés”, en donde se indicaba el nivel de estrés al que se sometía el cultivo en un punto del día.

## Capa de Presentación

La generación de visualizaciones se realizó a través de un portal web que permite consultar las lecturas obtenidas. Las representaciones mostradas ahí representan 3 enfoques organizacionales:

1. El nivel operacional está representado con una tabla donde se muestran las ultimas 20 lecturas, pues al observar los detalles, el área operativa puede tomar decisiones al momento.
2. El nivel gerencial muestra el comportamiento del día a través de una gráfica lineal que representa los valores y la hora en la que estos fueron registrados.
3. EL nivel directivo se ve representado a través de la gráfica de un indicador generado de la relación entre la humedad y temperatura del ambiente en las plantas, al que se le llama “Estrés Hídrico”.

Para graficar este último se utilizó una herramienta enfocada en inteligencia de negocio llamada **Tableau**, que permitió reducir tiempo en la creación de esta visualización y enfocarse en la generación teórica del indicador.

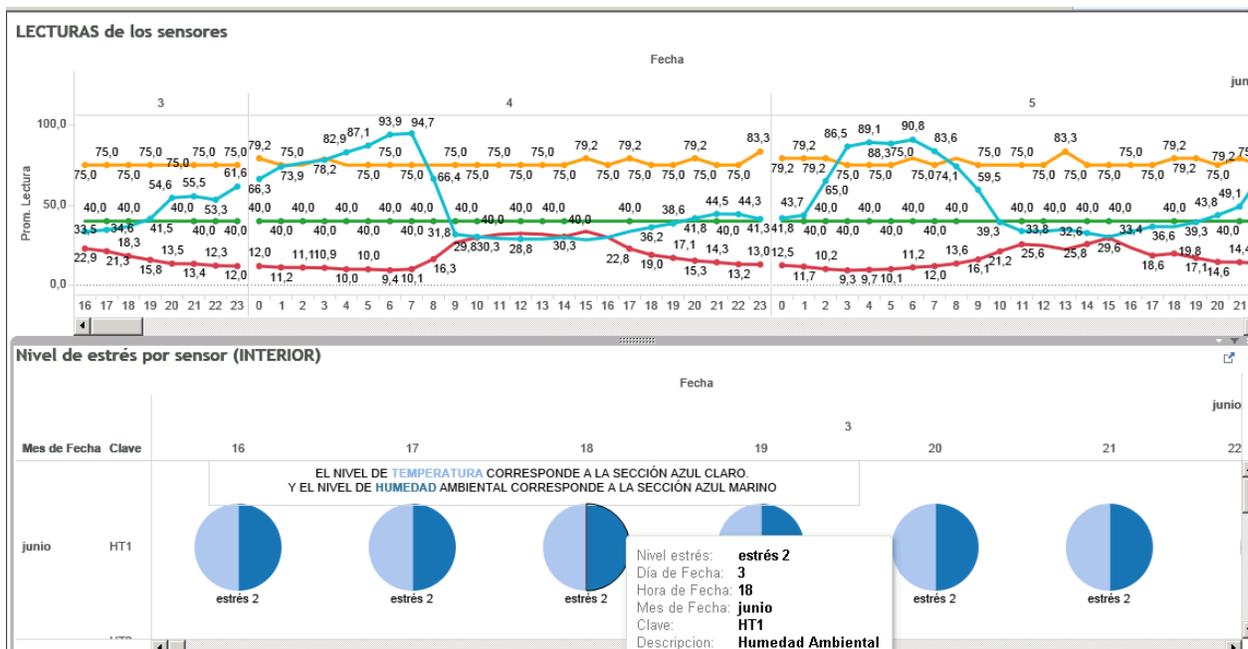


Ilustración 14: Una de las visualizaciones generadas con Tableau.

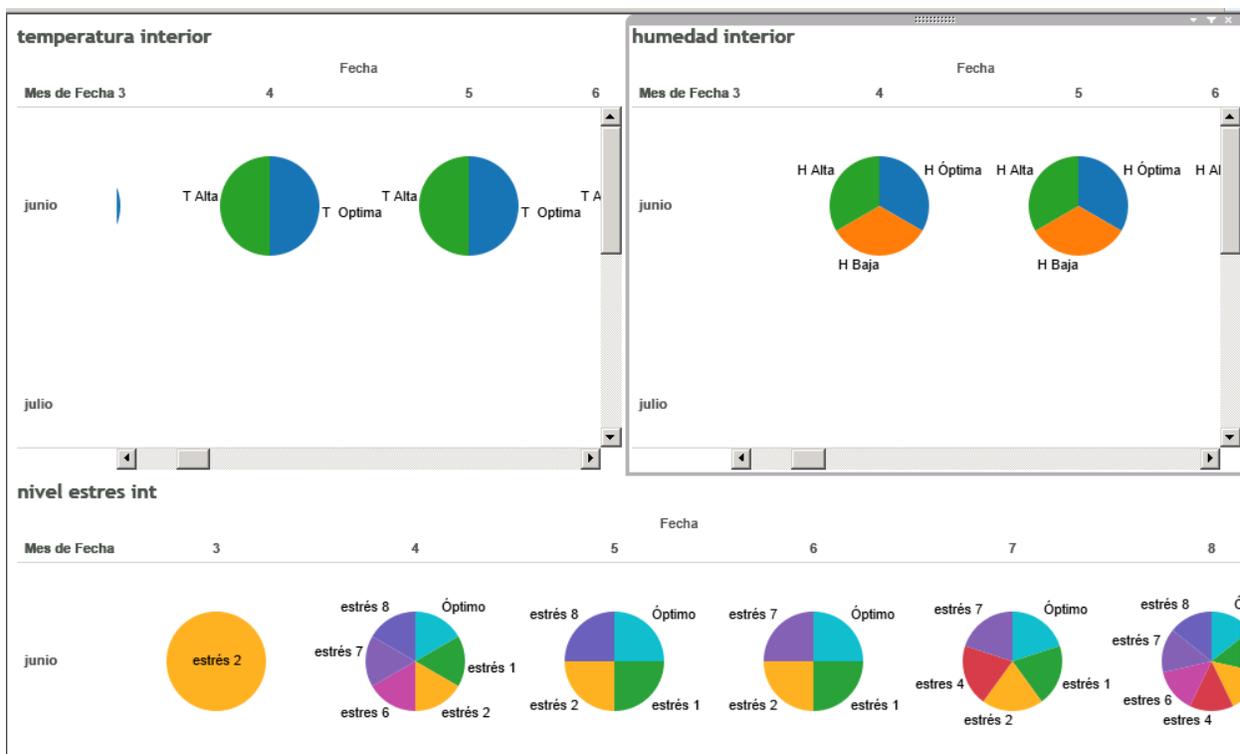


Ilustración 15: Otra de las visualizaciones generadas con Tableau, mostrando las lecturas de sensores por separado y el indicado de estrés hídrico..

## Soluciones Alternativas

En la actualidad existen algunos fabricantes que comercializan sensores especializados en las necesidades del campo. Sin embargo, para su funcionamiento es necesario adquirir además una consola que tome las lecturas de dichos sensores (La cual suele tener el doble o triple del costo de los sensores) lo cual los hace dependientes de una marca.

Un caso como el anterior podría ser de la marca BlueLab, empresa dedicada a la comercialización de sensores y consolas en actividades como agricultura, el cultivo hidropónico o los invernaderos, para cuyos nichos tienen una amplia variedad de productos. Su desventaja es que no son compatibles entre dispositivos similares, además de que para darles mantenimiento se vuelve el usuario dependiente de las refacciones de la misma marca.

Otra característica importante es la falta de conectividad entre estos dispositivos. Las consolas admiten la lectura de los sensores, pero debe realizarse in situ, por lo que para transmitir la información es necesario apoyarse en otros medios.

Algunas otras marcas como HACH, que se enfocan en aplicaciones más empresariales tienen algunas opciones para dotar de conectividad sus equipos. Tal es el caso de las tarjetas Modbus para sus lectoras que ofrecen más características, pero generalmente son módulos separados que tienen costos similares a los del resto de los equipos.

## Resultados

La primera prueba de concepto comenzó con una implementación controlada dentro de un domicilio, en donde se pusieron a prueba las capacidades de los sensores y la placa Arduino: Los sensores estuvieron en ambientes simulados como contenedores con mucha humedad, refrigeradores y la intemperie, macetas con diferentes humedades, líquidos con PH variable. Los resultados obtenidos de esta prueba de concepto sugirieron el rediseño de algunos componentes como funciones internas y estructura de datos, sobre lo cual se trabajó para el siguiente incremento.

Después de implementado el prototipo y una vez en marcha en el invernadero, se comenzó a poblar nuestra base de datos día a día, lo que nos permitió corroborar las palabras de la dueña del invernadero sobre las características climatológicas del lugar: Las variaciones de temperatura se podían producir en lapsos de 20 a 30 min, la humedad podía aumentar o disminuir debido a las lluvias, las madrugadas son muy frías y las tardes muy calurosas.

El prototipo sigue en funcionamiento y el objetivo de demostrar la aplicación práctica del modelo de referencia se cumplió.

La prueba de concepto fue **satisfactoria**. Entre los planes a mediano plazo se pretende generar un sistema que permita añadir funciones como alertas, clasificación por cultivo y una propuesta de proceso para gestionar la rotación de cultivos, su distribución, etc.

## **PC2: Herramienta de apoyo para sistemas de inventarios sobre líquidos**

El manejo de inventarios es una de las primeras herramientas que se adquieren cuando hablamos de nuevas organizaciones o sucursales de ya existentes, pues descuidar los bienes que se comercian (Sean productos o servicios) es un desliz que no se puede permitir si se desea prosperar.

En la actualidad existe un abanico de opciones en lo que a sistemas de inventarios se refiere: sistemas incluidos en ERPs, en puntos de venta, para almacenes especializados o muy genéricos, tiendas de conveniencia, activos fijos, etc. Sin embargo, hay un nicho que no ha sido del todo atendido, este es el de los líquidos.

Si bien es cierto que en la actualidad existen algunos sistemas de administración de líquidos, éstos no son del todo sencillos de implementar, pues implican cambios a los procesos y, en algunos casos a los hábitos mismos de las personas. Un error muy frecuente al implementar sistemas como ERPs es la pretensión de que el proceso se adapte al software cuando el enfoque debería ser el inverso.

### **Capa de negocio y proceso**

Se requiere la generación de un prototipo que integre hardware y software con un enfoque de Internet de las Cosas, que sirva como herramienta de apoyo en el proceso de gestión de inventarios para organizaciones en donde los líquidos representan la principal materia prima, siendo un auxiliar en las actividades de extracción y cuantificación de líquidos.

Otra ventaja de esta herramienta consistirá en una mejora al proceso de extracción, pues independientemente de los envases o contenedores a utilizar, solo se extraerá lo que los sensores indican (considerando la precisión de los mismos). De esta manera se implementaría una “medida de estandarización” en un tema en donde lograrlo puede ser complicado por la variedad de envases de plástico que se manejan para transportar los productos.

Por otro lado, permitirá agilizar el proceso de monetización de dichas extracciones, es decir, permitirá gestionar la compra o venta de dichos líquidos a través de una interfaz directa con sistemas de inventarios que manejen en una base de datos externa al sistema, de manera que pueda trabajar sin problemas operativos y/o legales.

## Capa de Objetos

Debido a que la actividad más importante para este problema es la de extraer líquido, se necesita conocer la cantidad exacta de líquido que se extrae de los contenedores del almacén. Así mismo debe existir un mecanismo que permita dejar pasar o parar el flujo del líquido, asegurando que solo se extraiga la cantidad necesaria del mismo.

## Capa de Sensores

Para esta prueba de concepto se utilizaron 3 tipos de dispositivos:

1. Un sensor de flujo para líquidos (o caudalímetro) que permite precisar el volumen que ha pasado a través de él en un determinado lapso. Consiste en una rueda y un activador, que se pone en operación gracias a que la rueda gira con el paso del agua, cada lectura representa un volumen determinado de líquido.
2. Una válvula solenoide (que mediante la presencia de voltaje realiza una acción, en este caso, abrirse) lo que permite el paso de líquido a través de mangueras.
3. Un relevador, el cual es un dispositivo (y actuador) que permite activarse y desactivarse cada que se aplica cierto voltaje.

Para mayor detalle de los dispositivos, consultar la documentación oficial de los distribuidores<sup>7</sup>.

## Capa de Pre-Procesamiento

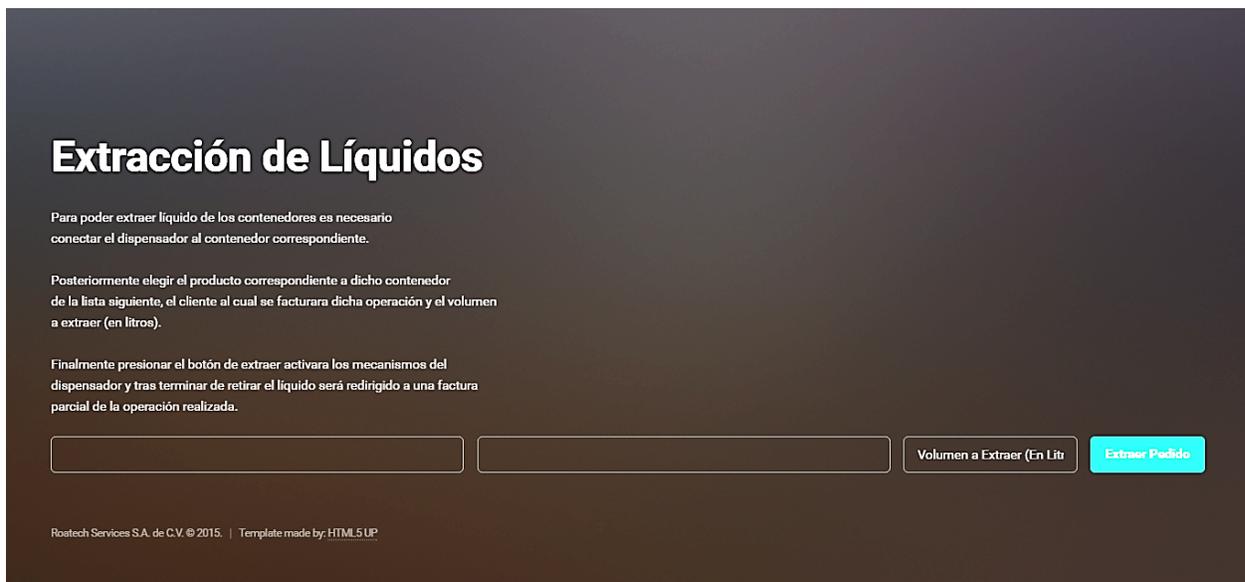
Para la realización de este prototipo se contó con una placa Arduino Yún, la cual contiene un microprocesador con un distro Linux que permite aplicaciones más complejas y robustas. Gracias a esto, la placa cumple con dos papeles importantes: como microcontrolador de los dispositivos y como servidor web con el que el usuario tiene interacción.

1. Su función como servidor web permite acceder al usuario y especificar los datos necesarios para comenzar la extracción de un producto.

---

<sup>7</sup> Sobre la documentación de los dispositivos corresponde a la siguiente: Caudalímetro <https://goo.gl/9Uc7kd>, solenoide <https://goo.gl/qcVmF8> y relevadores <https://goo.gl/gG6rdG>.

- a. Los usuarios acceden a la dirección IP o al nombre de la placa Yún, y esta los dirige a la página principal correspondiente, alojadas en una tarjeta SD en la placa.
  - b. Este portal permite consultar los productos líquidos de los que se tenga registro en el sistema de inventarios, así como a sus existencias, claves internas y otros datos relevantes. También tiene acceso a otros datos como los clientes, proveedores, compras realizadas, etc.
  - c. El usuario debe elegir un producto de la lista, un cliente a quien facturarle y una cantidad a extraer, que debe ser menor o igual a las existencias actuales.
  - d. Finalmente, tras una validación de los datos proporcionados por el usuario, se procede a enviarlos del servidor web al código de la placa, haciendo uso de mecanismos internos que permiten encapsular dichos datos y comunicarlos de manera transparente.
2. Una vez terminada la interacción con el usuario, los datos proporcionados son comunicados al microcontrolador que coordina el funcionamiento de los sensores.
- a. Una vez que los datos fueron enviados a la sección del microcontrolador, éste la toma y los extrae del encapsulamiento.
  - b. Debido a que el caudalímetro es un sensor que reacciona en “tiempo real” (Tan real como lo permite el ruido y las características intrínsecas de una señal analógica) se utiliza una función de la placa Arduino que permite realizar o “enganchar” una acción a un evento. En este caso, se “engancha” una función que aumente un contador cada vez que se detecte el paso en el mecanismo interno del caudalímetro, y posteriormente se calcule el volumen ya extraído.
  - c. En seguida se da la señal de activar el relevador, que permite fluir la corriente necesaria para abrir la válvula solenoide y permitir el flujo del líquido elegido.
  - d. Gracias al evento y su función, el líquido será extraído hasta que sea igual o superior al flujo indicado en la página web. En cualquiera de los casos, este volumen es identificado con una precisión de +- 3ml.
  - e. Una vez terminada la extracción, se cerrará la válvula, se realizan los últimos cálculos y se preparan los datos para el envío.



*Ilustración 16: Interfaz web para seleccionar los datos del líquido a extraer.*

El detalle de este flujo puede consultarse en la sección de especificación de casos de uso dentro de los anexos.

## Capa de Comunicaciones

A pesar de que la comunicación entre las secciones de la placa se realiza con una organización similar a la de un web service REST, la comunicación con la red se da en el momento en que se necesita comunicar los datos obtenidos con el sistema de inventarios.

Entonces la placa tiene también la función de cliente de un servidor destino en el cual se encuentran los mecanismos de recepción y almacenaje de datos. Para poder cumplir esta tarea es necesario haber terminado la conexión del usuario en el servidor web de la placa antes de poder utilizarla como cliente y enviar los datos, esto con el fin de no saturar la capacidad de la placa. Este detalle se debe tener en cuenta cuando se habla de concurrencia.

Los datos importantes son: la clave del producto, la clave del cliente a facturar y el volumen extraído, que se convierten a un objeto con formato JSON y se envían a través de Internet al servidor destino. Una vez más para esto nos valemos del protocolo TCP, usando el método POST en un paquete usando el estándar HTTP 1.1.

## Capa de Almacenamiento y procesamiento

Debido a que utilizamos la base de datos de un sistema externo para almacenar los datos del líquido extraído, las tareas de almacenaje y procesamiento son ejecutadas al mismo nivel pues hay que ejecutar una serie de inserciones en las tablas de la base de datos a modo de que el sistema de inventarios reconozca la extracción como válida y evite conflictos posteriores.

En el ambiente del servidor ocurre una interacción más sencilla que en la prueba anterior.

- ~ Una vez recibido el paquete de datos por parte del Arduino, se recupera la información sobre el producto y volumen extraído.
- ~ Después se envían a un procedimiento almacenado en la base de datos, el cual se encarga de las inserciones pertinentes, pues para que el sistema reconozca dichos datos como correctos son necesarias la actualización e inserción consecutiva a diferentes tablas. Para lo que utilizamos el concepto de transacciones.
  1. Se complementa los datos de entrada al obtener otros como el nombre del cliente, del producto, su precio, y se da formato a las claves a utilizar.
  2. Se registra en bitácora de movimientos, con el fin de generar la evidencia de que se realizará.
  3. Se registra en movimientos de almacén el volumen a extraer, para que este no sea tomado en cuenta si se realizara alguna consulta en ese momento.
  4. Posteriormente se realiza el registro de la factura, con los datos de encabezado: cliente, total, impuestos, fecha, etc.
  5. Con el registro anterior se genera el detalle: el producto, el volumen, precio unitario, monto y total.
  6. Se actualiza la bitácora con el número de documento (Factura) y su total.
  7. Se actualizan también las existencias en el inventario.

Finalmente se actualizan los folios de facturas, para que vayan acorde a lo insertado.

## **Capa de Presentación**

En este caso, la generación de visualizaciones va más enfocada a cómo generar reportes y quizás cubos de información que aporten más valor a los usuarios de lo que lo harían los reportes generados nativamente por la aplicación.

Por otro lado, gracias a que sistemas como SAE permiten la inclusión de campos abiertos en sus registros, se abre un sinfín de posibilidades al utilizar herramientas de inteligencia de negocio. Por ejemplo: ¿Que tan útil sería para el área de contabilidad si pudiéramos decirles los consumos que se han realizado desde inventario con tan solo clasificarlos por las partidas a las que se asocian dichos consumos? ¿U observar el comportamiento de nuestras ventas en tiempo real o un resumen de corte dinámico?

## **Soluciones Alternativas**

Actualmente en el mercado existen aplicaciones que permiten la administración e inventario de productos líquidos para empresas que manejen este tipo de productos. Sin embargo, estas soluciones suelen incluir cambios a veces innecesarios en los procesos de trabajo.

Además de que dichas soluciones se ven influenciadas por otras variables de la organización como las medidas que emplean y la forma en la que se calcula el volumen a utilizar. Si se tienen herramientas irregulares o no estandarizadas, esto puede llevar a datos erróneos e información imprecisa.

## **Resultados**

La segunda prueba de concepto se terminó como un prototipo en un ambiente controlado, esto es, se verificó su funcionamiento y comportamiento en diferentes situaciones. Sin embargo, por una situación suscitada dentro de la organización donde se implementaría el prototipo, su implementación fue aplazada y, por cuestiones de tiempo, no fue incluida dentro del documento.

No obstante, esto abre las posibilidades de mostrar cómo se pueden generar herramientas de apoyo similares para sistemas en otras áreas de una organización y que, a su vez, complementando esta información pueda aportar valor en áreas relacionadas: ¿Qué pasaría si pudiéramos generar los

pedidos a proveedores en automático cuando las existencias sean menores a las permitidas por almacenes, y tanto adquisiciones como contabilidad pudieran recibir solo la actualización del lote y su compromiso respectivamente, mejorando la relación y facilitando la comunicación entre esas y otras áreas? Con esta prueba se demuestra que no solo las “cosas” que pueden conectarse al IoT sean sistemas a la medida, dispositivos y objetos presentes en la nube, podemos también añadir los softwares comerciales y enfocarlo a un segmento tan delicado de cuando cambios se trata como lo son las organizaciones.

## Conclusiones

El modelo presentado en este trabajo propone la organización de soluciones en donde sus elementos (Que incluyen hardware, software y herramientas de negocio) son organizados a través de capas con objetivos y responsabilidades definidos que se comunican entre sí, con el objetivo de ofrecer productos y servicios al aplicar una tecnología emergente, como lo es el Internet de las Cosas, al área de negocios.

A partir de esta organización es posible definir una arquitectura de sistema para la solución planteada que permita la documentación y el apego a estándares que rigen el área de TI, y con esto generar un valor agregado en comparación con el enfoque utilizado por la corriente “maker”, la cual persigue la generación de soluciones basado en un pensamiento de “hágalo usted mismo” y que reduce los productos generados a inventos para una necesidad personal que dependen de la retroalimentación de una comunidad para su mejora continua.

Las pruebas de concepto presentadas prueban la viabilidad del método al dar como resultado dos productos tangibles: Uno de ellos implementado en el lugar de trabajo y el otro desarrollado a un nivel de prototipo funcional que permiten demostrar flexibilidad para trabajar con diferentes tipos de tecnologías, según la problemática a resolver, así como los beneficios que se obtienen al organizar la solución resultante en las capas propuestas y poder analizar, diseñar e implementar como un todo.

Sin embargo, de esta primera implementación del modelo también se derivan una serie de lecciones aprendidas que se traducen en mejoras a su definición, como parte de un proceso de mejora y madurez, una de las principales es la manera en la cual se puede conformar una especificación particular del método que sea repetible, es decir, elegir un conjunto de herramientas y tecnologías para trabajar por cada capa y definir sus vías de comunicación para compartir datos de entrada y salida entre ellas. De esta manera dicha especificación podría facilitar la difusión y el entendimiento del método sin necesidad de alejarse de su naturaleza adaptable.

Cabe destacar la importancia de que tecnologías emergentes como el Internet de las Cosas sean añadidas al conjunto de herramientas que el estudiante en informática dispone para la generación de soluciones que tienen como objetivo apoyar la toma de decisiones en una organización, ya que como bien lo mencionan la descripción de la carrera y el perfil del egresado en informática, debe de

adaptarse a los avances tecnológicos, pero siempre haciendo uso de las mejores prácticas y técnicas para su labor. Porque en una época donde los teléfonos inteligentes se han vuelto parte de nuestro día a día, y vivimos el auge de dispositivos interconectados entre sí, una visión donde los sistemas de información son conformados solamente por software es demasiado cerrada.

## Anexos

### PC1 – Documento de Visión.

#### Sistema de Monitoreo para Invernaderos

##### Visión

### Introducción

El propósito de este documento consiste en la definición conceptual del proyecto para un sistema de monitoreo de variables para invernaderos. Compuesto principalmente de tres secciones, en las cuales se pretende hablar, de manera muy general, de la problemática a la cual responde la generación de este proyecto, los actores principales que intervienen en el mismo, así como el ambiente en el que se desenvuelven y finalmente de las características más representativas que debe contener el producto final, esto con el fin de delimitar el dominio del problema al cual se atacara.

### Posicionamiento

#### Definición de problemática

El problema de	La falta de un proceso de seguimiento establecido para el funcionamiento de un invernadero, aunado a los altos costos de las soluciones que permiten medir ciertas variables en los mismos.
Afecta a	A los dueños de pequeños y medianos invernaderos cuyo propósito es el generar de forma sustentable productos agrícolas para consumo propio.
El impacto de este es	Un alto costo de mantenimiento debido al uso exclusivo de técnicas empíricas, así como a una aceleración del desgaste natural en la tierra dedicada a dicha actividad.
Una solución adecuada podría ser	Un sistema de monitoreo de variables de bajo costo que, utilizando hardware y software libre, genere un ecosistema que apoye las técnicas actualmente utilizadas en el campo y aumente su calidad.

## Posicionamiento del producto

Para	Los dueños de parcelas o invernaderos autosustentables
Quienes	Realizan el cuidado de sus siembras a través de las técnicas conocidas sin el apoyo de herramientas tecnológicas
El sistema de monitoreo para invernaderos	Es un producto integrador
Que	Permite el monitoreo de variables que afectan a los cultivos y cuya consulta puede realizarse en cualquier momento
A diferencia de	Los dispositivos actuales que permiten la medición de variables, los cuales necesitan lectores especializados, no tienen conectividad a internet y tienen un costo alto
Nuestro producto	Se basa en componentes libres y permite la visualización en línea a través de gráficas y tablas informativas.

## Descripción de Stakeholders

### Resumen de Stakeholders

Nombre	Descripción	Responsabilidades
Sponsor	<ul style="list-style-type: none"> <li>Principal fuente de financiamiento para el proyecto.</li> </ul>	<ul style="list-style-type: none"> <li>Las decisiones sobre las propuestas para la adquisición de materiales y dispositivos recaen principalmente sobre él.</li> <li>Así mismo tiene un gran peso en decisiones relacionadas a la planeación y ejecución del proyecto.</li> </ul>
Experto en procesos	<ul style="list-style-type: none"> <li>Provee de asesoramiento sobre procesos de negocio.</li> </ul>	<ul style="list-style-type: none"> <li>Principalmente apoya en la comprensión del proceso de negocio actual, con el fin de poder alinear la solución al mismo, generando valor al usuario con el producto.</li> </ul>

Nombre	Descripción	Responsabilidades
Usuario experto	<ul style="list-style-type: none"> <li>• Principal proveedor de información debido al dominio de los procesos de producción.</li> <li>• Principal interesado en la funcionalidad del sistema.</li> </ul>	<ul style="list-style-type: none"> <li>• Es el encargado de proveer de información sobre los procesos de trabajo, así como de aclarar dudas que puedan surgir durante el desarrollo.</li> <li>• Así mismo es el principal aportador de requerimientos para el sistema, así como fungir como la principal instancia para la discusión y modificación de los mismos.</li> </ul>
Usuario	<ul style="list-style-type: none"> <li>• Principal usuario del sistema.</li> <li>• Puede tratarse de personal de apoyo en los procesos de producción.</li> </ul>	<ul style="list-style-type: none"> <li>• Se limita al consumo de información generada por el sistema.</li> <li>• Puede proveer de algunos requerimientos, los cuales mayoritariamente serán no funcionales.</li> </ul>

## Ambiente de Usuario

El invernadero donde se desarrollará el Sistema está localizado en la delegación Tlalpan, en las cercanías con el cerro del Ajusco. El invernadero en cuestión representa una extensión de tierra con aproximadamente 239.25m<sup>2</sup>, dividida en dos partes: La primera es una zona externa y techada por una malla de tela, en donde se encuentran entre 6 ~ 8 parcelas (Depende la temporada) al ras del suelo, donde pueden cultivarse papas, rábanos, zanahorias, fresas, entre otros. La segunda zona consiste en una zona cubierta de plástico blanco, que permite una mayor protección ante las temperaturas y agentes ambientales (Los cuales son un tanto extremos debido a la localización del lugar) y en las que los cultivos están principalmente en macetas, donde podemos ver hierbas como perejil o cilantro, u otros cultivos como los ajos.

Este invernadero surgió como resultado de una convocatoria por parte del gobierno federal para apoyar la creación y desarrollo de invernaderos autosustentables en la ciudad de México, en esta época el lugar era operado por la dueña del mismo y auxiliada por dos personas que eran subcontratadas con el apoyo antes mencionado, sin embargo, cuando este expiro la operación recayó exclusivamente en la Srita. Escalante, quien debido a sus

obligaciones solo puede dedicar un día a la semana a su mantenimiento (Lunes, para ser exactos) por lo que el resto de la semana este queda “descuidado”.

Por otro lado, las actividades de siembra, mantenimiento y cosecha de los cultivos son realizadas con técnicas empíricas, es decir gracias a la experiencia (propia o externa) lo que, aunado a los periodos entre mantenimientos, representa una baja en la productividad e incluso en la calidad de los productos, así como un aumento en los costos de mantenimiento, pues la inversión para salvar las cosechas suele ser mayor.

## Resumen del Producto

### Necesidades y Características

Necesidad	Prioridad	Características	Fecha de lanzamiento o estimada
Se requiere la medición constante de las siguientes variables que afectan el crecimiento de los cultivos del invernadero: <ul style="list-style-type: none"> <li>• Humedad y temperatura ambiental.</li> <li>• Humedad de superficie.</li> <li>• pH de superficie.</li> </ul>	Alta	<ul style="list-style-type: none"> <li>• La lectura de sensores debe hacerse cada 10 min.</li> <li>• Se deberán distribuir sensores tanto en la parte externa como en la interna del terreno.</li> </ul>	03/06/2015
Se requiere que estas lecturas sean enviadas a un servidor en la nube y almacenadas en una base de datos.	Alta	<ul style="list-style-type: none"> <li>• Las lecturas debes ser enviadas en conjunto.</li> <li>• Las lecturas deben ser enviadas en formato JSON.</li> </ul>	05/06/2015

Se requiere que las lecturas puedan ser visualizadas desde una interfaz gráfica web, con el fin de mostrar el estado actual de dichas variables en el invernadero.	Alta	<ul style="list-style-type: none"> <li>Las lecturas almacenadas deben ser mostradas en gráficos y tablas representativas.</li> <li>Los valores deben ser mostrados como porcentajes o valores reales, según sea el caso.</li> </ul>	10/06/2015
--	------	---	------------

### Otros Requerimientos

Requerimiento	Prioridad	Fecha de lanzamiento planeada
Se requiere que el sistema de monitoreo sea de bajo costo	Alta	
Se requiere que los sensores de algunas variables puedan cambiar de lugar.	Alta	
Se requiere que la interfaz gráfica web pueda ser mostrada en diferentes dispositivos.	Baja	

## **PC1 – Plan de Proyecto**

### **Sistema de Monitoreo para Invernaderos Plan de Proyecto**

#### **Introducción**

El propósito de este documento consiste en exponer y detallar el plan de acción para la generación del Sistema de monitoreo para invernaderos, el cual consiste en una sección sobre el personal que estará colaborando en su construcción, así como las guías, buenas prácticas y lineamientos de los que estos se apoyaran para la construcción del producto.

Así mismo se detallan los principales objetivos e hitos del proyecto, así como la referencia a los casos de uso y riesgos relacionados a ellos, con lo que se pretende evaluar el progreso del proyecto. También se incluye una sección con una descripción sobre el plan de despliegue para producto final y una sección dedicada a las lecciones aprendidas durante el desarrollo del mismo, con el fin de auxiliar a la mejora continua del grupo de trabajo.

#### **Organización del Proyecto**

El sistema de monitoreo de invernaderos representa la primera prueba de concepto de un modelo por capas que propone la incursión del internet de las cosas en sistemas administrativos (Y no por eso necesariamente informáticos) por lo que actualmente solo trabaja un estudiante de informática en la UNAM, el tesista Ayrton Daniel Ruiz Reyes desempeñando el papel de analista, diseñador, programador, tester, implementador y documentador del mismo.

#### **Estándares, Métricas y Practicas del Proyecto**

Para el desarrollo del sistema de monitoreo de variables para invernaderos se optó por utilizar la metodología de trabajo OpenUp, la cual consiste en un marco de trabajo ágil e iterativo para el desarrollo de software. Si bien es cierto que parte del proyecto consiste en el desarrollo de un módulo de consulta, también es cierto que la parte de hardware que integra actualmente se considera como un prototipo, por lo que la naturaleza iterativa de OpenUp beneficia la mejora continua de este.

Otro de las herramientas a utilizar será el lenguaje de modelado unificado (o UML por sus siglas en inglés) el cual es un estándar en desarrollo de sistemas utilizado para representar gráficamente las características, funcionalidades, organización entre otras características de un desarrollo de manera gráfica y fácil de entender, auxiliado por el software StarUML, el cual se apega al estándar UML 2.0.

En la parte de hardware nos auxiliaremos de la herramienta Fritzing la cual permite diseñar gráficamente la conexión entre diversos componentes, incluyendo productos Arduino, facilitando la documentación de la parte física del Sistema.

## Hitos y Objetivos del Proyecto

Iteración	Objetivos primarios (riesgos y escenarios en casos de uso)	Fecha de planeación o Hito	Duración estimada
I1	<ol style="list-style-type: none"> <li>Obtener las principales características del producto a desarrollar. (Especificación de requerimientos generales de sistema y Lista de riesgos).</li> <li>Conocer el ambiente del usuario y la situación actual de la organización. (Hito: Documento de Visión).</li> <li>Con la información recabada presentar los casos de uso más importantes. (Hito: Modelo de casos de uso).</li> </ol>	01/05/2015 – 8/05/2015	6 días
E1	<ol style="list-style-type: none"> <li>Especificar la arquitectura con la que se construirá el producto, basada en el modelo de referencia de una investigación realizada. (Hito: Cuaderno de arquitectura).</li> <li>Especificar los casos de uso detectados en el modelo de casos de uso (Hito: Especificación de casos de uso).</li> </ol>	11/05/2015 – 15/05/2015	5 días
C1	<ol style="list-style-type: none"> <li>Desarrollar el modulo del comportamiento en el microcontrolador.</li> <li>Desarrollar los módulos de recepción de datos y visualización a nivel servidor.</li> </ol>	18/05/2015 – 29/05/2015	10 días
C2	<ol style="list-style-type: none"> <li>Especificar la manera en la que se realizaran las pruebas de laboratorio sobre el prototipo desarrollado</li> <li>Realizar pruebas de integración en un ambiente controlado.</li> </ol>	27/05/2015 – 29/05/2015	3 días
T1	<ol style="list-style-type: none"> <li>Especificar la manera en la que se realizará la entrega e instalación del prototipo desarrollado (Hito: Plan de despliegue).</li> <li>Realizar las pruebas finales de comportamiento.</li> </ol>	01/06/2015 – 12/06/2015	10 días

## Despliegue

El despliegue del prototipo estará dividido en tres etapas:

1. La primera será una liberación parcial en un ambiente controlado, con el fin de probar algunas características de los dispositivos ante ciertos estímulos extremos: Su resistencia ante un trabajo constante, ante temperaturas o escenarios extremos, etc. Una vez terminado el periodo de prueba, será mostrado al usuario, con quien se retroalimentará al equipo con el fin de detallar el prototipo.
2. Posteriormente se realizará una liberación completa, siendo esta ya la instalación de los dispositivos y las adecuaciones necesarias en el entorno para su instalación.
3. Finalmente, ya con el modelo instalado y en funcionamiento, tras permitirnos la recolección de datos se pasará a una última retroalimentación con el usuario a modo de identificar relaciones, indicadores y, en general, información útil para el apoyo de su operación.

## Lecciones Aprendidas

- Debido a que se está trabajando con un paradigma relativamente reciente, la tarea de definir parámetros que permitan indicar el estado actual del proyecto se vuelve una tarea más demandante que en el desarrollo exclusivo de sistemas.
- A pesar de que, para el desarrollo de prototipos, se suele estimar una iteración por fase, no hay que perder de vista el hecho de que dicho prototipo se encuentra en constante actualización, es decir, en constante iteración con cada una de las fases del proyecto.

## **PC1 - Especificación General de Requerimientos para el Sistema**

### **Sistema de Monitoreo para Invernaderos Especificación General de Requerimientos para el Sistema**

#### **Introducción**

El propósito de este documento consiste en presentar información sobre características generales del sistema, así como las cualidades que deben cumplir. El documento se divide en tres secciones principales: La primera abarca los requerimientos de sistema y las características que tienen que cumplir para satisfacer las necesidades de los usuarios, La segunda se centra en las interfaces que tendrá el sistema, abarcando desde las gráficas y la distribución de pantallas, hasta las necesarias para tener interacción con otros sistemas, sean estos programas externos o dispositivos físicos. Finalmente, la tercera parte abarca requerimientos administrativos, entre los que se destacan los lineamientos legales a los que se apegará el sistema a desarrollar y la documentación que acompaña al producto final.

Cabe destacar la importancia de este documento pues funge de medio para concentrar requerimientos y características de diferentes áreas en un lugar común que permita una fácil lectura y una rápida consulta en fases más avanzadas del desarrollo.

#### **Requerimientos generales de funcionalidad**

- ~ El sistema permitirá la obtención de mediciones de los sensores conectados a él.
- ~ Los sensores disponibles miden cuatro variables: La temperatura y humedad ambiental, la humedad de superficie y el pH del suelo.
- ~ El sistema enviará las lecturas a una nube privada alojada fuera de la red local.
- ~ El sistema mostrará las lecturas a través de un portal web.
- ~ El portal web mostrará gráficas generadas de las lecturas recibidas.

## **Cualidades del Sistema**

### **Usabilidad**

- ~ El prototipo debe de permitir el cambio de locación de los sensores de humedad de superficie entre las diferentes macetas.
- ~ El prototipo debe permitir el cambio de locación del sensor de pH de suelo entre las diferentes macetas.
- ~ La información mostrada en la graficas del portal web debe ser consistente con los datos alojados en la nube privada.
- ~ La información mostrada en la graficas deben ser representativas y de fácil lectura para el usuario.

### **Confiabilidad**

- ~ El prototipo debe ser capaz de operar en un ambiente con suministro eléctrico variable.
- ~ El prototipo debe ser capaz de operar en ambientes con altos niveles de humedad.

### **Desempeño**

- ~ Los sensores de temperatura y humedad ambiental deben de ser leídos cada 10 minutos.
- ~ Los sensores de humedad de superficie deben ser medido cada 5 min.
- ~ Los sensores de pH de suelo deben ser medidos cada 30 min.

### **Compatibilidad**

- ~ El portal web debe ser compatible con los navegadores Google Chrome, Mozilla Firefox, Safari y derivados.
- ~ El portal web debe de desplegarse de manera correcta en dispositivos móviles.

## **Interfaces de sistema**

### **Interfaces de usuario**

#### *Look & Feel*

- ~ Debido al reducido número de elementos que se desplegarán en la pantalla, se ha optado por un diseño sencillo y minimalista, que permita una interfaz sencilla pero limpia, que consista en 3 secciones para contenido principalmente, las cuales contendrán una estructura con las últimas lecturas, un gráfico de las lecturas del día y finalmente una sección donde se incluirán gráficos del BI.
- ~ Este representa el diseño inicial del prototipo, sin embargo, nuevas secciones serán añadidas en posteriores versiones.

#### *Requerimientos de diseño y navegación*

No aplica

#### *Consistencia*

- ~ Como el alcance de la primera versión del producto fue delimitada a enfocarse en recibir datos solamente, la interfaz está diseñada con el único propósito de mostrar datos, es por eso que no se utilizan páginas internas u otras secciones además de la principal. Esto no significa que no se tengan contempladas otras funciones que serían añadidas en etapas posteriores.

#### *Requerimientos de personalización y adaptación a usuarios*

No aplica

## **Interfaces a sistemas externos y/o dispositivos**

#### *Interfaces de software*

- ~ Debido a la arquitectura tecnológica con la que se está trabajando, el producto tendrá interacción con un software para inteligencia de negocios, en este caso Tableau, con el cual se explotarán los datos recibidos de los sensores, permitiendo generar visualizaciones atractivas y de fácil entendimiento para el usuario final. Para hacer uso de dicho software es necesario generar una conexión a la base de datos donde residen las lecturas, y una vez que

las visualizaciones están construidas estas pueden ser incrustadas utilizando etiquetas HTML.

#### *Interfaces de hardware*

- ~ El producto se comunica con un microcontrolador, dispositivo encargado de manejar los sensores, de recibir sus lecturas y de enviarlas a la nube. Dicho manejo se da a través de señales eléctricas con diversas características, las cuales son traducidas por el microcontrolador en valores numéricos.

#### *Interfaces de comunicación*

- ~ Para que la comunicación entre la nube y los sensores se efectúe, se realiza una conexión a través de internet utilizando el protocolo HTTP, el cual permite el envío de mensajes en cuyo contenido se encuentran las lecturas correspondientes a un momento del día, estas están en un formato JSON, garantizando una compatibilidad en lenguajes.

### **Reglas de Negocio**

- ~ Debido a que en una primera etapa el sistema se enfoca en recabar y presentar lecturas del entorno, estas no están sujetas a reglas de negocio en el estricto sentido del concepto. Sin embargo, debido a la mejora continua y retroalimentación obtenida es una posibilidad que se plantea para etapas posteriores.

### **Restricciones de Sistema**

- ~ El producto debe ser de bajo costo.

### **Requisitos de Conformidad para el sistema**

#### **Requerimientos de licencia**

- ~ Los componentes del producto están basados en tecnologías de código abierto, las cuales están protegidas por licencias de libre uso y distribución.

### **Notas legales, derechos de autor y otros**

No aplica

### **Estándares aplicados**

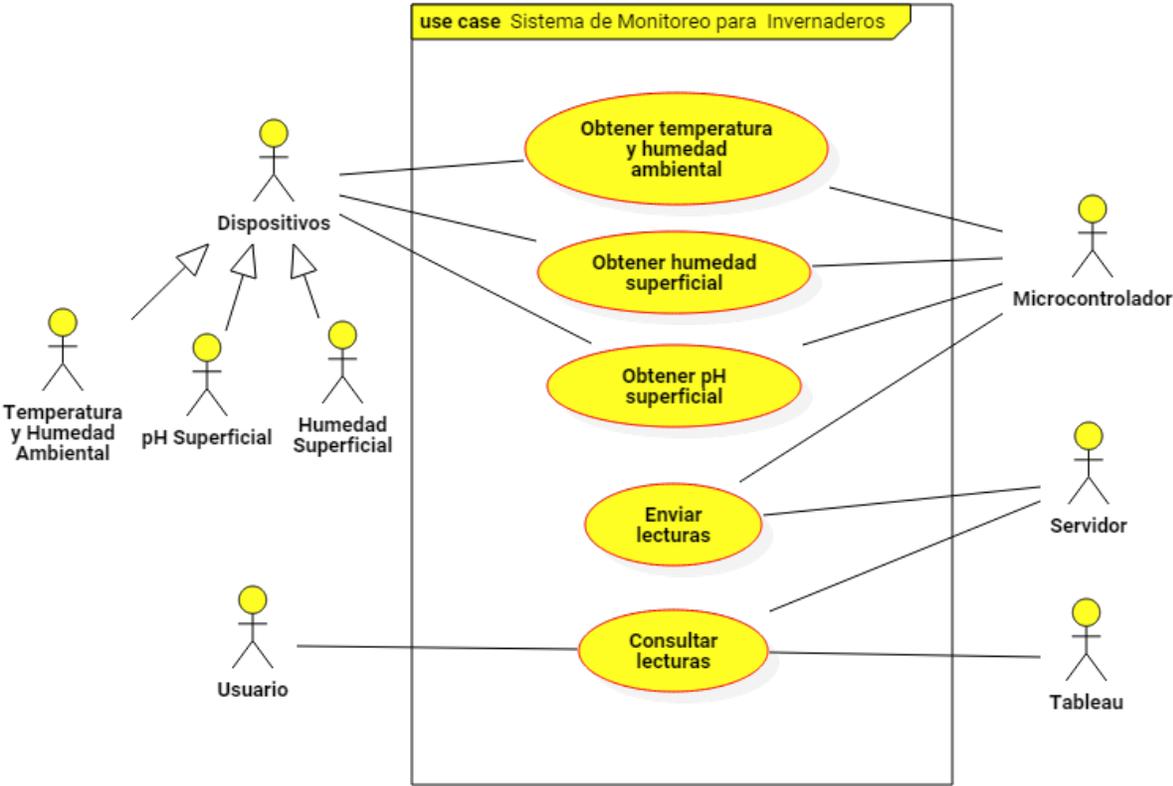
Cada uno de los elementos del diseño y desarrollo del producto están basados en estándares que rigen la industria:

- ~ La administración del proyecto se lleva a través de la metodología OpenUP, auxiliada de algunas buenas practicas por parte del PMI.
- ~ Los diagramas del diseño se encuentran basados en el estándar UML
- ~ Los dispositivos electrónicos como el microcontrolador y los sensores cumplen con estándares de funcionalidad de la IEEE y de calidad ISO9000.
- ~ El diseño y construcción de la base de datos se apega a las reglas de normalización y al estándar ANSI-SQL.
- ~ El desarrollo de la lógica del microcontrolador se realiza con lenguaje C++, el cual tiene como base el estándar ANSI-C.
- ~ Las comunicaciones entre partes del producto se llevan a cabo apegados al estándar TCP/IP y a los protocolos HTTP y JSON.
- ~ Para el desarrollo web se utiliza el estándar HTML5, CSS y JS, así como guías de usabilidad y diseño de sitios web.

## **Documentación del Sistema**

La documentación del producto abarca meramente los documentos y evidencia que respalden el diseño y modelado del sistema.

PC1 – Diagrama de Casos de Uso



## PC1 – Especificación de Casos de Uso

### Sistema de Monitoreo para Invernaderos Especificación de Casos de Uso

#### Caso de Uso: Obtener Temperatura y Humedad Ambiental

##### Breve Descripción

El sistema obtendrá la lectura correspondiente a los sensores de temperatura y humedad ambiental, así como realizará el tratamiento necesario para traducir dicha lectura.

##### Descripción de Actores

###### *Sensor de Temperatura y Humedad Ambiental*

Sensor digital que permite obtener la temperatura a través de un termistor y la humedad con un conjunto de resistencias montadas en un único dispositivo.

###### *Microcontrolador*

Consiste en una placa de circuitería que concentra la conexión entre dispositivos compatibles, controlada por un microcontrolador re-grabable que permite guardar instrucciones sobre el comportamiento y tratamiento de sus diferentes componentes.

##### Precondiciones

- ~ El sistema conozca el número total de sensores de temperatura y humedad ambiental se encuentran conectados.
- ~ El sistema conozca el número de pines en los que están conectados los sensores.

##### Flujo de Eventos Básico

1. El caso de uso comienza cuando el sistema detecta que han transcurrido 10 minutos.
2. El sistema le indica al microcontrolador que lea el sensor ubicado en un pin en particular.
3. El sistema recibe el valor obtenido del microcontrolador, realiza la conversión de temperatura a grados centígrados y de humedad a porcentaje.
4. El sistema realiza el paso dos y tres por cada sensor que se tenga conectado en los diferentes pines conocidos.
5. El sistema almacena los resultados de las lecturas en una estructura de datos.

6. Incluye referencia al caso de uso "Enviar Lecturas".
7. El caso de uso termina finaliza.

### **Flujos Alternativos**

*Código de error durante la lectura de los sensores.*

Si durante el punto 4 del flujo principal, el microcontrolador devuelve un código de error como respuesta del sensor, entonces:

1. El sistema volverá a obtener una respuesta por parte del sensor.
2. Si el error persiste 2 ocasiones más, dicho sensor será omitido hasta el próximo bloque de lectura.
3. El caso de uso continúa en el paso 5.

### **Post-Condiciones**

*Registro en base de datos*

Al término del caso de uso se puede encontrar en la base de datos un nuevo registro por sensor consultado, donde figura la lectura, el sensor y tipo al que corresponde, así como la fecha y hora sobre la que se ingresó.

### **Requerimientos Especiales**

- ~ Los números correspondientes a los pines de los sensores deben pertenecer a los destinados para señales digitales.

## Caso de Uso: Obtener Humedad Superficial

### Breve Descripción

El sistema obtendrá la lectura correspondiente a los sensores de humedad superficial incrustada en la tierra, así como realizará el tratamiento necesario para traducir dicha lectura.

### Descripción de Actores

#### *Sensor de Humedad Superficial*

Sensor análogo que permite obtener el grado de humedad en una superficie terrosa, a través de la reacción de un componente químico localizado en las probetas del dicho sensor.

#### *Microcontrolador*

Consiste en una placa de circuitería que concentra la conexión entre dispositivos compatibles, controlada por un microcontrolador re-grabable que permite guardar instrucciones sobre el comportamiento y tratamiento de sus diferentes componentes.

### Precondiciones

- ~ El sistema conozca el número total de sensores de humedad superficial se encuentran conectados.
- ~ El sistema conozca el número de pines en los que están conectados los sensores.
- ~ El sistema tiene definido un intervalo de valores de referencia para la traducción del valor recibido por el sensor en porcentaje.

### Flujo de Eventos Básico

1. El caso de uso comienza cuando el sistema detecta que han transcurrido 5 minutos.
2. El sistema le indica al microcontrolador que lea el sensor ubicado en un pin en particular.
3. El sistema recibe el valor obtenido del microcontrolador, realiza la comparación con un intervalo definido como parámetro de referencia y asigna el porcentaje correspondiente.
4. El sistema realiza el paso dos y tres por cada sensor que se tenga conectado en los diferentes pines conocidos.
5. El sistema almacena los resultados de las lecturas en una estructura de datos.
6. Incluye referencia al caso de uso "Enviar Lecturas".

7. El caso de uso termina finaliza.

### **Flujos Alternativos**

*Valor del sensor fuera del intervalo de referencia.*

Si durante el punto 4 del flujo principal, el microcontrolador devuelve un valor fuera del rango de referencia conocido, entonces:

1. El sistema mapeara el valor obtenido entre el intervalo conocido, a modo de obtener una traducción consistente con los porcentajes utilizados.
2. El caso de uso continúa en el paso 5.

### **Post-Condiciones**

*Registro en base de datos*

Al término del caso de uso se puede encontrar en la base de datos un nuevo registro por sensor consultado, donde figura la lectura, el sensor y tipo al que corresponde, así como la fecha y hora sobre la que se ingresó.

### **Requerimientos Especiales**

- ~ Los números de pines para los sensores deben de coincidir con los destinados a señales análogas.
- ~ Los sensores deben encontrarse enterrados dentro de la tierra.

## Caso de Uso: Obtener pH Superficial

### Breve Descripción

El sistema obtendrá la lectura correspondiente al pH superficial colocado en la tierra, así como realizará el tratamiento necesario para traducir dicha lectura.

### Descripción de Actores

#### *Sensor de pH Superficial*

Sensor análogo que permite obtener el grado acidez en la tierra, parámetro de importancia según el tipo de cultivo a germinar pues determina los nutrientes de dicha sección de tierra. Esta medida la realiza gracias a una calibración previamente aplicada que permite definir un intervalo de acidez en que concentrarse.

#### *Microcontrolador*

Consiste en una placa de circuitería que concentra la conexión entre dispositivos compatibles, controlada por un microcontrolador re-grabable que permite guardar instrucciones sobre el comportamiento y tratamiento de sus diferentes componentes.

### Precondiciones

- ~ El sensor de pH ha realizado su calibración necesaria, definiendo el intervalo de acidez a medir.
- ~ El sistema conozco el número de pin al cual está conectado el sensor.
- ~ El sistema tiene definido el intervalo de acidez a la que corresponde la calibración del sensor.

### Flujo de Eventos Básico

1. El caso de uso comienza cuando el sistema detecta que han transcurrido 30 minutos.
2. El sistema le indica al microcontrolador que lea el sensor ubicado en un pin en particular.
3. El sistema recibe el valor obtenido del microcontrolador, realiza la comparación con el intervalo de acidez conocido y asigna el valor correspondiente.
4. El sistema almacena el resultado de la lectura en una estructura de datos.
5. Incluye referencia al caso de uso "Enviar Lecturas".
6. El caso de uso termina finaliza.

## **Flujos Alternativos**

*Valor del sensor fuera del intervalo de acidez definido.*

Si durante el punto 3 del flujo principal, el microcontrolador devuelve un valor fuera del rango de acidez conocido, entonces:

1. El sistema mapeara el valor obtenido entre el intervalo conocido, a modo de obtener una traducción consistente con los valores calibrados.
2. El caso de uso continúa en el paso 5.

## **Post-Condiciones**

*Registro en base de datos*

Al término del caso de uso se puede encontrar en la base de datos un nuevo registro por sensor consultado, donde figura la lectura, el sensor y tipo al que corresponde, así como la fecha y hora sobre la que se ingresó.

## **Requerimientos Especiales**

- ~ El número de pin debe coincidir con el intervalo destinado a señales análogas.
- ~ El sensor debe encontrarse enterrado en la tierra para poder generar mediciones consistentes.

## Caso de Uso: Enviar lecturas

### Breve Descripción

El sistema enviara las lecturas obtenidas de los sensores a través de Internet a una infraestructura privada localizada a kilómetros de distancia del prototipo, en donde dichas lecturas serán guardadas y explotadas.

### Descripción de Actores

#### *Nube Privada*

Se refiere a una infraestructura localizada a kilómetros de distancia, la cual se compone de un servidor configurado con las utilidades necesarias para obtener las lecturas enviadas, para guardarlas en una base de datos asegurando su persistencia, que cuenta con las rutinas necesarias para procesar dichos datos y ofrecer información de calidad para el usuario final.

#### *Microcontrolador*

Consiste en una placa de circuitería que concentra la conexión entre dispositivos compatibles, controlada por un microcontrolador re-grabable que permite guardar instrucciones sobre el comportamiento y tratamiento de sus diferentes componentes.

### Precondiciones

- ~ Las lecturas deben estar almacenadas en una estructura de datos dentro del microcontrolador.
- ~ El sistema conoce la dirección IP que tiene la nube privada.

### Flujo de Eventos Básico

1. El caso de uso comienza recibiendo una estructura de datos definida donde se localizan las lecturas recabadas.
2. El sistema traduce la estructura de datos a una con formato JSON.
3. El sistema indica al microcontrolador que genere una conexión a través de Internet a la dirección de la nube privada.
4. La nube privada responde que la conexión fue exitosa.
5. El sistema indica al microcontrolador que envíe la estructura JSON al servidor, a través de la conexión establecida previamente.

6. La nube privada responde que la recepción del envío fue correcto y termina la conexión.
7. El sistema obtiene de la estructura JSON recibida una lectura enviada.
8. El sistema inserta en la base de datos la lectura obtenida.
9. Los pasos 7 y 8 se repiten mientras existan lecturas restantes en la estructura recibida.
10. El caso de uso termina.

## **Flujos Alternativos**

### *Fallo en creación de conexión para Internet*

Si durante el punto 3 del flujo principal, el microcontrolador devuelve un error en la generación de la conexión, entonces:

1. El sistema reintentara la conexión con la nube privada.
2. El caso de uso continúa en el paso 5.
3. Si el caso se repite tres veces, entonces se desechan las lecturas de la estructura.

### *Fallo con el envío de la estructura JSON sobre Internet*

Si durante el punto 5 del flujo principal, el microcontrolador devuelve un error en el envío de la estructura JSON, entonces:

1. El sistema reintentara el envío de la estructura a la nube privada.
2. El caso de uso continúa en el paso 6.
3. Si el caso se repite tres veces, entonces se desechan las lecturas de la estructura.

## **Post-Condiciones**

### *Registro en bitácoras de control*

Al término del caso de uso se puede encontrar un registro con los datos correspondientes al prototipo en la bitácora de conexiones en la nube de datos.

## **Requerimientos Especiales**

*No aplica*

## **Caso de Uso: Consulta Lecturas**

### **Breve Descripción**

El sistema mostrara al usuario un resumen de las lecturas recibidas, incluyendo gráficos y visualizaciones utilizando un software de inteligencia de negocio, el cual permite realizar explotación de datos enfocada a las necesidades de los usuarios

### **Descripción de Actores**

#### *Nube Privada*

Se refiere a una infraestructura localizada a kilómetros de distancia, la cual se compone de un servidor configurado con las utilidades necesarias para obtener las lecturas enviadas, para guardarlas en una base de datos asegurando su persistencia, que cuenta con las rutinas necesarias para procesar dichos datos y ofrecer información de calidad para el usuario final.

#### *Tableau*

Se trata de un software de inteligencia de negocios que permite el análisis de datos provenientes de fuentes diversas con el fin de obtener información de valor para los usuarios, pero a través de procesos minuciosos y específicos que dan como resultado un paradigma de dimensiones con múltiples posibles cruces de datos, brindando nuevas perspectivas de los mismos.

#### *Usuario*

Se refiere a las personas que trabajan en el Invernadero, las cuales están interesadas en conocer que información puede ser ofrecida de las lecturas procesadas por el sistema.

### **Precondiciones**

- ~ El sistema debe estar activo en la nube privada, es decir, el administrador de aplicaciones debe estar ejecutándose.
- ~ La nube de datos debe permitir que se acceda a dicha aplicación, teniendo habilitada un acceso público.

### **Flujo de Eventos Básico**

1. El caso de uso comienza cuando el usuario accede al sitio web montado en la nube privada.
2. El sistema realizara una consulta a la base de datos obteniendo los últimos 20 resultados recibidos del Invernadero.

3. El sistema graficara las lecturas del día actual del invernadero.
4. El sistema realizara una petición al software Tableau para que pueda acceder a los reportes generados en la plataforma.
5. El sistema mostrara dichos elementos en pantalla.
6. El caso de uso termina finaliza.

## **Flujos Alternativos**

### *Sitio web inaccesible*

Si durante el punto 1 del flujo principal, existiera algún problema con el sitio web, entonces:

1. El sistema mostrara una pantalla de error, informando que en estos momentos se realiza una tarea de mantenimiento.
2. El caso de uso continúa en el paso 6.

### *Base de datos no disponible*

Si durante el punto 2 y 3 del flujo principal, existiera algún problema durante la conexión con la base de datos, entonces:

1. El sistema mostrara en las secciones pertinentes un botón para volver a obtener los datos.
2. El usuario presiona el botón.
3. El sistema muestra una pantalla de espera mientras intenta conectar con la base de datos.
4. El caso de uso continúa en el paso 3.

### *Sesión expirada en Tableau*

Si durante el punto 1 del flujo principal, existiera algún problema con el sitio web, entonces

1. El sistema mostrara una pantalla de error, informando que en estos momentos se realiza una tarea de mantenimiento.
2. El caso de uso continúa en el paso 6.

## **Post-Condiciones**

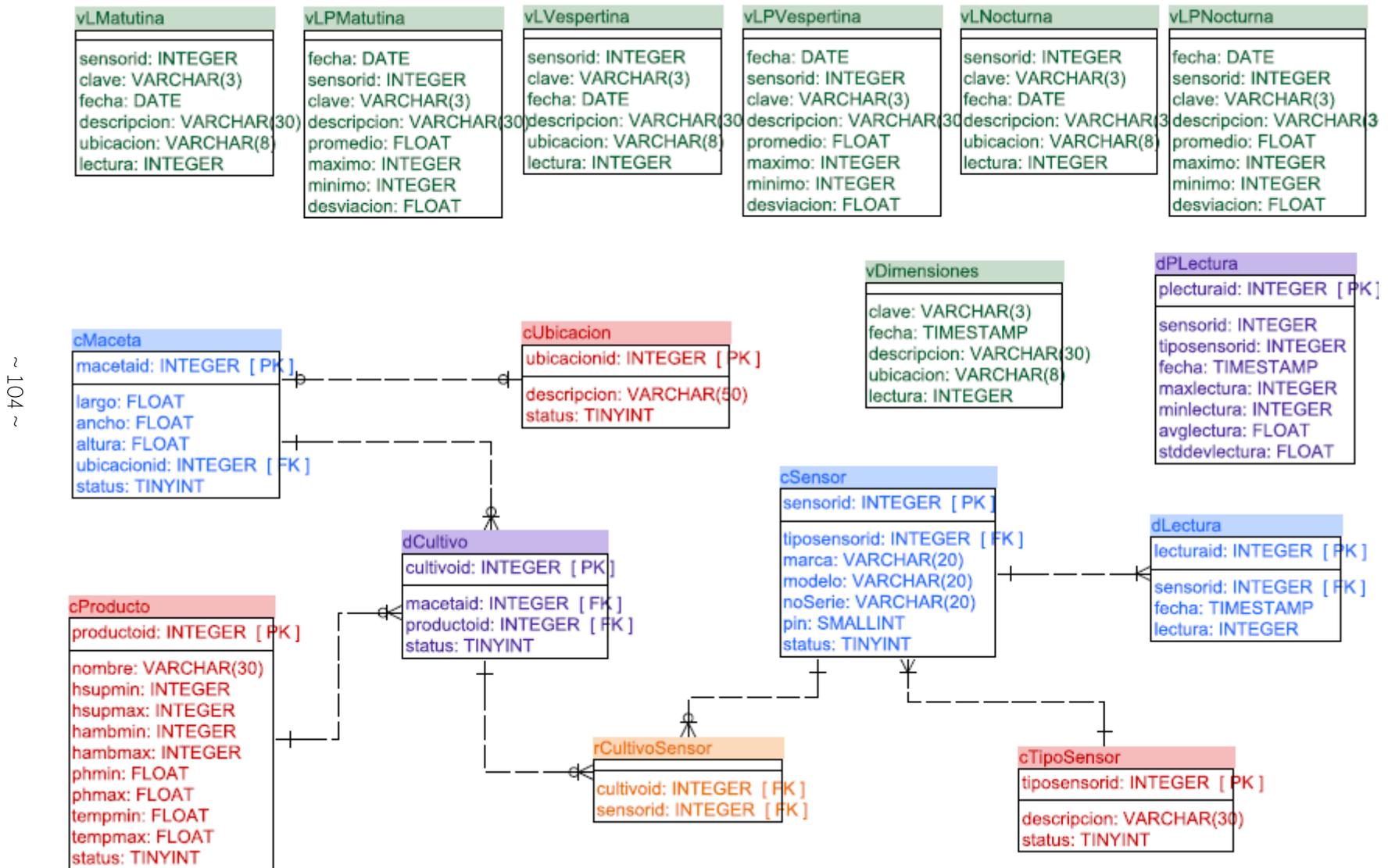
### *Registro en base de datos*

Al término del caso de uso se puede encontrar en la base de datos un nuevo registro por sensor consultado, donde figura la lectura, el sensor y tipo al que corresponde, así como la fecha y hora sobre la que se ingresó.

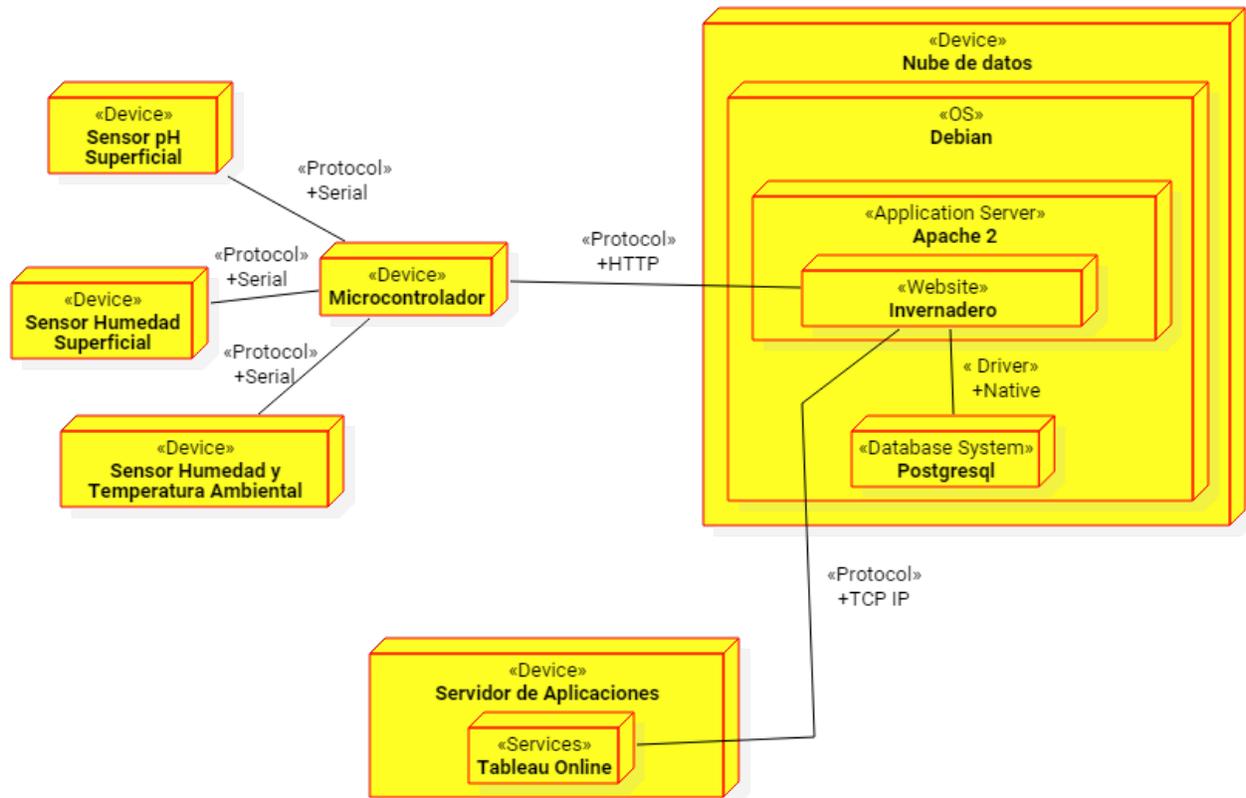
## **Requerimientos Especiales**

Se requiere acceso a los servicios de Tableau a través de su suite Server o en su defecto del servicio online.

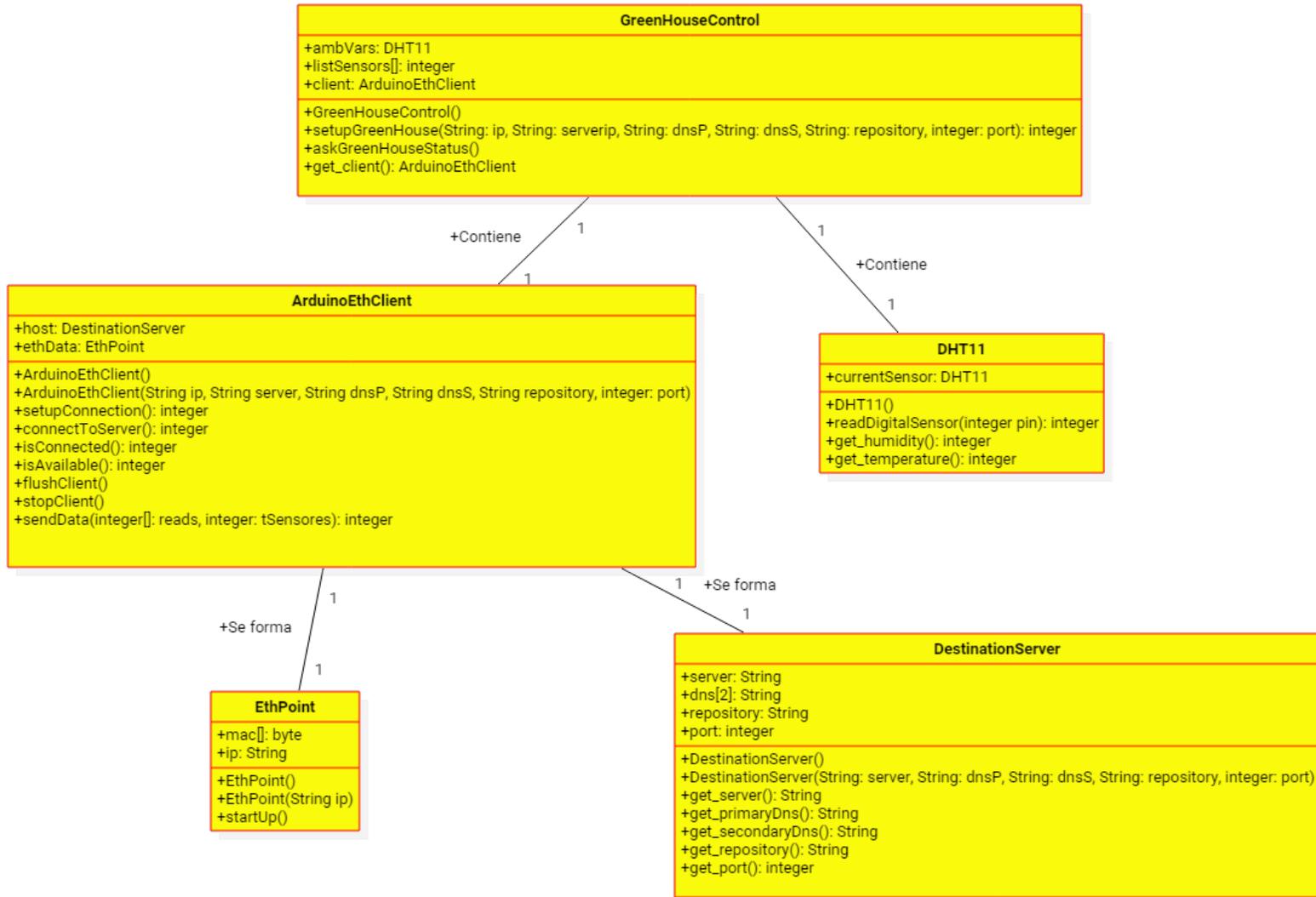
## PC1 – Diagrama de Entidad Relación



# PC1 – Diagrama de Componentes



## PC1 – Diagrama de Clases



## PC1 – Extracto de Código

### Archivo Invernadero\_2.0\_ino

```
#include <SPI.h>
#include <Ethernet.h>
#include <Dns.h>
#include <ArduinoJson.h>
#include <Event.h>
#include <Timer.h>
#include <dht.h>
#include "GreenHouseControl.h"

GreenHouseControl ghc;

void setup() {
    int status = 0;
    Serial.begin(9600);
    status = ghc.setupGreenHouse(IPAddress(192,168,1,59), IPAddress(201,124,199,31),
    IPAddress(200,33,148,156),IPAddress(148,240,118,40), "/Invernadero/", 91);
    unsigned long lapseTime = 10 * (60 * 1000);
    status = ghc.get_manager().every(2000, askStatus, 0);
}

void loop() {
    ghc.get_manager().update();
}

void askStatus(void *context){
    ghc.askGreenHouseStatus();
}
```

## Archivo GreenHouseControl.h

```
#ifndef GreenHouseControl_h
#define GreenHouseControl_h

#include <Event.h>
#include <Timer.h>
#include <Ethernet.h>
#include <EthernetClient.h>
#include <EthernetServer.h>
#include <ArduinoJson.h>
#include "ArduinoEthClient.h"
#include "DHT11.h"

#define TOTALNUMSENSOR 8

class GreenHouseControl
{
private:
    ArduinoEthClient greenHouseClient;
    Timer taskManager;
    DHT11 ambVars;
    static int listSensor[];
    EthernetClient client;
public:
    GreenHouseControl();
    int setupGreenHouse(IPAddress ip, IPAddress serverip, IPAddress dnsP, IPAddress dnsS,
        String repository, int port);
    void askGreenHouseStatus();
    Timer& get_manager();
    ArduinoEthClient& get_client ();
};
#endif
```

## Archivo GreenHouseControl.cpp

```
#include "GreenHouseControl.h"

int GreenHouseControl::listSensor[] = {7,8,9,14,15,16,17,18};

GreenHouseControl::GreenHouseControl(){}

int GreenHouseControl::setupGreenHouse(IPAddress ip, IPAddress serverip, IPAddress dnsP,
IPAddress dnsS, String repository, int port){

    GreenHouseControl::greenHouseClient = ArduinoEthClient(ip, serverip, dnsP, dnsS,
repository, port);

    GreenHouseControl::greenHouseClient.setupConnection();

    GreenHouseControl::ambVars = DHT11();

    return 0;

}

void GreenHouseControl::askGreenHouseStatus(){

    int data[20];

    int y = 0, z = 0, sv = 0;

    StaticJsonBuffer<560> jsonBuffer;

    JsonObject& root = jsonBuffer.createObject();

    JsonArray& datas = root.createNestedArray("D");

    for (int i = 0; z < TOTALNUMSENSOR; i++) {

        y = GreenHouseControl::listSensor[z];

        data[i] = y;

        if (y == 7 || y == 8 || y == 9) {

            GreenHouseControl::ambVars.readDigitalSensor(y);

            JsonObject& nObj = datas.createNestedObject();

            nObj["p"] = y;

            nObj["h"] = GreenHouseControl::ambVars.get_humidity();

            nObj["t"] = GreenHouseControl::ambVars.get_temperature();

        } else if (y == 14) {

            JsonObject& nObjAn = datas.createNestedObject();

            sv = analogRead(y);

            nObjAn["p"] = y;

            nObjAn["l"] = sv;

        } else {

            JsonObject& nObjAn = datas.createNestedObject();
```

```

        sv = analogRead(y);
        nObjAn["p"] = y;
        nObjAn["l"] = sv;
    }
z++;
}
char buffer[550];
root.printTo(buffer, sizeof(buffer));
Serial.println(buffer);
client.flush();
IPAddress ip(201,124,199,31);
if (client.connect(ip, 91)){
    Serial.print("conectado");
    String request = "GET /Invernadero/recieveReads.php?data=";
    request.concat(buffer);
    Serial.print(request);
    client.print(request);
    client.print(" HTTP/1.1");
    client.println("Host: roatech.dyndns.info");
    client.println("User-Agent: ArduinoEth/1.1");
    client.println("Connection: close");
    client.println();
    client.stop();
} else {
    Serial.println("1");
}
}

Timer& GreenHouseControl::get_manager() {
    return GreenHouseControl::taskManager;
}

ArduinoEthClient& GreenHouseControl::get_client() {
    return GreenHouseControl::greenHouseClient;
}

```

**PC1 – Memoria Fotográfica**



*Ilustración 17: Parte externa del Invernadero.*



*Ilustración 18: Parte interna del invernadero.*



*Ilustración 19: La caja contenedora fue impresa en 3D, por lo que está diseñada para acomodar el microcontrolador y la distribución de los sensores.*



*Ilustración 20: Organización final del cableado en el microcontrolador.*



Ilustración 21: Sensores de humedad y temperatura ambiental (DHT 11).



*Ilustración 22: Sensores de humedad de superficie los cuales se entierran en la tierra (izquierda), y están equipados con metros extra de cable que permiten moverlos entre macetas (derecha).*



*Ilustración 23: Sensor de pH de suelo, el cual se entierra en la tierra para obtener lecturas.*

## PC2 – Documento de Visión

### Sistema de Inventario para Líquidos Visión

#### Introducción

El propósito de este documento consiste en la definición conceptual del proyecto para un sistema de inventarios para el control de líquidos. Compuesto principalmente de tres secciones, en las cuales se pretende hablar, de manera muy general, de la problemática a la cual responde la generación de este proyecto, los actores principales que intervienen en el mismo, así como el ambiente en el que se desenvuelven y finalmente de las características más representativas que debe contener el producto final, esto con el fin de delimitar el dominio del problema al cual se atacara.

#### Posicionamiento

##### Definición de problemática

El problema de	La falta de implementación de medidas estandarizadas para la extracción de productos en las empresas cuyo principal insumo son líquidos químicos
Afecta a	A los dueños de pequeñas y medianas empresas que distribuyen dichos líquidos, ya sea como producto o como materia prima
El impacto de este es	Un alto riesgo de error durante la creación de sus productos a partir de diferentes líquidos, aunado al robo hormiga, que resulta en un costo de merma desconocido
Una solución adecuada podría ser	Un sistema de inventario de líquidos que permita regular la extracción, venta y movimiento de los mismos dentro del almacén, sirviendo como vía de comunicación a los sistemas de inventario que se utilicen en dicha empresa.

## Posicionamiento del producto

Para	Los dueños de empresas cuyo principal insumo de trabajo sean productos líquidos
Quienes	Realizan la venta y distribución de dichos líquidos, ya sea como materias primas o como un producto final.
El sistema de monitoreo para invernaderos	Es un producto integrador
Que	Permite mantener controlado el inventario, extracción y movimiento de sus insumos entre el área de trabajo y los sistemas auxiliares que utilicen
A diferencia de	Los sistemas de inventario convencionales que no contemplan este tipo de empresas.
Nuestro producto	Se basa en componentes libres y estándares que permiten la interconexión con sistemas externos, garantizando el uso de los mismos, con un enfoque innovador.

## Descripción de Stakeholders

### Resumen de Stakeholders

Nombre	Descripción	Responsabilidades
Sponsor	<ul style="list-style-type: none"> <li>Principal fuente de financiamiento para el proyecto.</li> </ul>	<ul style="list-style-type: none"> <li>Las decisiones sobre las propuestas para la adquisición de materiales y dispositivos recaen principalmente sobre él.</li> <li>Así mismo tiene un gran peso en decisiones relacionadas a la planeación y ejecución del proyecto.</li> </ul>
Experto en procesos	<ul style="list-style-type: none"> <li>Provee de asesoramiento sobre procesos de negocio.</li> </ul>	<ul style="list-style-type: none"> <li>Principalmente apoya en la comprensión del proceso de negocio actual, con el fin de poder alinear la solución al mismo, generando valor al usuario con el producto.</li> </ul>
Usuario experto	<ul style="list-style-type: none"> <li>Principal proveedor de información debido al dominio de los procesos de</li> </ul>	<ul style="list-style-type: none"> <li>Es el encargado de proveer de información sobre los procesos de trabajo, así como de aclarar dudas que puedan surgir durante el desarrollo.</li> </ul>

Nombre	Descripción	Responsabilidades
	producción. <ul style="list-style-type: none"> <li>• Principal interesado en la funcionalidad del sistema.</li> </ul>	<ul style="list-style-type: none"> <li>• Así mismo es el principal aportador de requerimientos para el sistema, así como fungir como la principal instancia para la discusión y modificación de los mismos.</li> </ul>
Usuario	<ul style="list-style-type: none"> <li>• Principal usuario del sistema.</li> <li>• Puede tratarse de personal de apoyo en los procesos de producción.</li> </ul>	<ul style="list-style-type: none"> <li>• Se limita al consumo de información generada por el sistema.</li> <li>• Puede proveer de algunos requerimientos, los cuales mayoritariamente serán no funcionales.</li> </ul>

## Ambiente de Usuario

La empresa Q3 S.A. de C.V. es una distribuidora de fragancias y productos de belleza con sede en la ciudad de México. Entre su catálogo de productos, ellos distribuyen químicos como materia prima a otros productores o bien, productos ya creados basados en formulas químicas desarrolladas por el especialista de su laboratorio.

Q3 se enfrenta a tres problemas fundamentales: El primero de ellos es la manera en que se extraen los químicos que van a ser vendidos, ya que si son para generar productos estos son extraídos en envases de productos comerciales que no suelen ser exactos en su volumen, ejemplo de ellos serían bote de yogurt de 1 kilo de producto, por el contrario, si se trata de venta como materia prima los envases en los que distribuyen son de plástico que en ocasiones es irregular, por lo que tres envases de 1 galón cada uno, pueden tener volumen más o volumen menos en realidad.

El segundo problema viene derivado del primero, pues por la falta de exactitud en las extracciones, este suele prestarse a el robo hormiga de cantidades pequeñas de producto, las cuales pueden sumar varios miles de pesos en su conjunto.

El tercer problema es el control interno de los líquidos, pues a cada extracción de volumen, este se registra en una bitácora manualmente, para ser ingresada a un sistema de inventarios al termino del día, este control suele no coincidir con el producto real contenido en las bodegas debido a las dos situaciones anteriores.

## Resumen del Producto

### Necesidades y Características

Necesidad	Prioridad	Características	Fecha de lanzamiento estimada
Se requiere de un dispositivo que permita cuantificar el volumen extraído de un líquido especificado a través de una interfaz web	Alta	<ul style="list-style-type: none"><li>La interfaz debe de contener el catálogo de productos del sistema de inventarios.</li><li>El volumen extraído debe ser igual al especificado en la interfaz web.</li></ul>	15/06/2015
Se requiere que los datos de la extracción sean registrados en el sistema de inventarios como una salida.	Alta	<ul style="list-style-type: none"><li>Los registros deben reflejar que fueron extraídos por el dispositivo.</li><li>Debe especificarse si fue por venta o para la generación de producto</li></ul>	23/06/2015

### Otros Requerimientos

Requerimiento	Prioridad	Fecha de lanzamiento planeada
Se requiere que el sistema conviva con el sistema de inventarios actual.	Alta	
Se requiere que el sistema trabaje dentro de la infraestructura interna de la empresa.	Alta	

## **PC2 - Especificación General de Requerimientos para el Sistema**

### **Sistema de Inventario para Líquidos Especificación General de Requerimientos para el Sistema**

#### **Introducción**

El propósito de este documento consiste en presentar información sobre características generales del sistema, así como las cualidades que deben cumplir. El documento se divide en tres secciones principales: La primera abarca los requerimientos de sistema y las características que tienen que cumplir para satisfacer las necesidades de los usuarios, La segunda se centra en las interfaces que tendrá el sistema, abarcando desde las gráficas y la distribución de pantallas, hasta las necesarias para tener interacción con otros sistemas, sean estos programas externos o dispositivos físicos. Finalmente, la tercera parte abarca requerimientos administrativos, entre los que se destacan los lineamientos legales a los que se apegará el sistema a desarrollar y la documentación que acompaña al producto final.

Cabe destacar la importancia de este documento pues funge de medio para concentrar requerimientos y características de diferentes áreas en un lugar común que permita una fácil lectura y una rápida consulta en fases más avanzadas del desarrollo.

#### **Requerimientos generales de funcionalidad**

- ~ El sistema permitirá especificar el producto y el volumen que se extraerá.
- ~ Se trabajará con una válvula solenoide y un caudalímetro para la medición del volumen.
- ~ El sistema envía sus lecturas a un servidor localizado en la red interna del lugar.
- ~ Las lecturas enviadas serán procesadas y almacenadas en la base de datos del sistema de inventarios actual.

## **Cualidades del Sistema**

### ***Usabilidad***

- ~ El prototipo debe de permitir que el dispositivo de medición pueda ser utilizado en diferentes contenedores de productos.
- ~ La información reflejada en el sistema de inventarios debe ser consistente con los datos especificados antes de extraer el volumen.

### **Confiabilidad**

- ~ El prototipo debe de tener una precisión de +- 20ml.
- ~ El prototipo debe ser capaz de operar en ambientes con altos niveles de humedad.
- ~ El prototipo debe tener una protección impermeable contra derrames de líquidos sobre él.

### **Desempeño**

- ~ El tiempo de respuesta entre la especificación del producto y volumen y su extracción debe ser mínimo,

### **Compatibilidad**

- ~ La interfaz web en la que se especifica el producto y volumen a extraer debe ser compatible con los equipos con los que se cuenta en la oficina.
- ~ El servidor que aloja el sistema de inventarios debe permitir al dispositivo conectarse a él, para enviarle los datos de la extracción.

## **Interfaces de sistema**

### **Interfaces de usuario**

#### *Look & Feel*

- ~ Debido al reducido número de elementos que se desplegaran en la pantalla, se ha optado por un diseño sencillo y minimalista, que permita una interfaz sencilla y limpia, que consista en 2 listados, uno de productos y uno de clientes, así como un campo para especificar el volumen a extraer.
- ~ Este representa el diseño inicial del prototipo, sin embargo, nuevas funcionalidades podrían ser añadidas en posteriores versiones.

#### *Requerimientos de diseño y navegación*

No aplica

#### *Consistencia*

- ~ Dentro de la interfaz donde se especifica el producto y volumen a extraer, se debe considerar que no puede extraerse más producto del que actualmente se tiene disponible en bodega.

#### *Requerimientos de personalización y adaptación a usuarios*

No aplica

## **Interfaces a sistemas externos y/o dispositivos**

### **Interfaces de software**

- ~ Debido a la arquitectura tecnológica con la que se está trabajando, el producto tendrá interacción con un sistema de inventarios externo, a través de su base de datos, a modo de evitar conflictos de funcionalidad y legales con los acuerdos de usuario final de dicho software.

### **Interfaces de hardware**

- ~ El producto coordinara la interfaz web que se encargara de proveer de los datos de extracción al sistema de inventarios, así como la interacción del microcontrolador con la válvula de paso y el caudalímetro.

## **Interfaces de comunicación**

- ~ Para que la comunicación entre el servidor y el dispositivo desarrollado se efectúe, se realizará una conexión a través de internet utilizando el protocolo HTTP, el cual permite el envío de mensajes en cuyo contenido se encuentran los datos del líquido y volumen a extraer en un formato JSON, garantizando su interpretación entre diferentes lenguajes de programación.

## **Reglas de Negocio**

- No puede extraerse más volumen del que el sistema de inventarios tiene en sus registros.
- En caso de que la extracción sea una venta, debe de especificarse el cliente al cual se le está despachando su pedido.

## **Restricciones de Sistema**

- ~ El producto debe ser de bajo costo, tanto en mantenimiento como en reparación

## **Requisitos de Conformidad para el sistema**

### **Requerimientos de licencia**

- ~ Los componentes del producto están basados en tecnologías de código abierto, las cuales están protegidas por licencias de libre uso y distribución.

### **Notas legales, derechos de autor y otros**

*No aplica*

### **Estándares aplicados**

Cada uno de los elementos del diseño y desarrollo del producto están basados en estándares que rigen la industria:

- ~ La administración del proyecto se lleva a través de la metodología OpenUP, auxiliada de algunas buenas practicas por parte del PMI.
- ~ Los diagramas del diseño se encuentran basados en el estándar UML
- ~ Los dispositivos electrónicos como el microcontrolador y los sensores cumplen con estándares de funcionalidad de la IEEE y de calidad ISO9000.
- ~ El desarrollo de la lógica del microcontrolador se realiza con lenguaje C++, el cual tiene como base el estándar ANSI-C.
- ~ Las comunicaciones entre partes del producto se llevan a cabo apegados al estándar TCP/IP

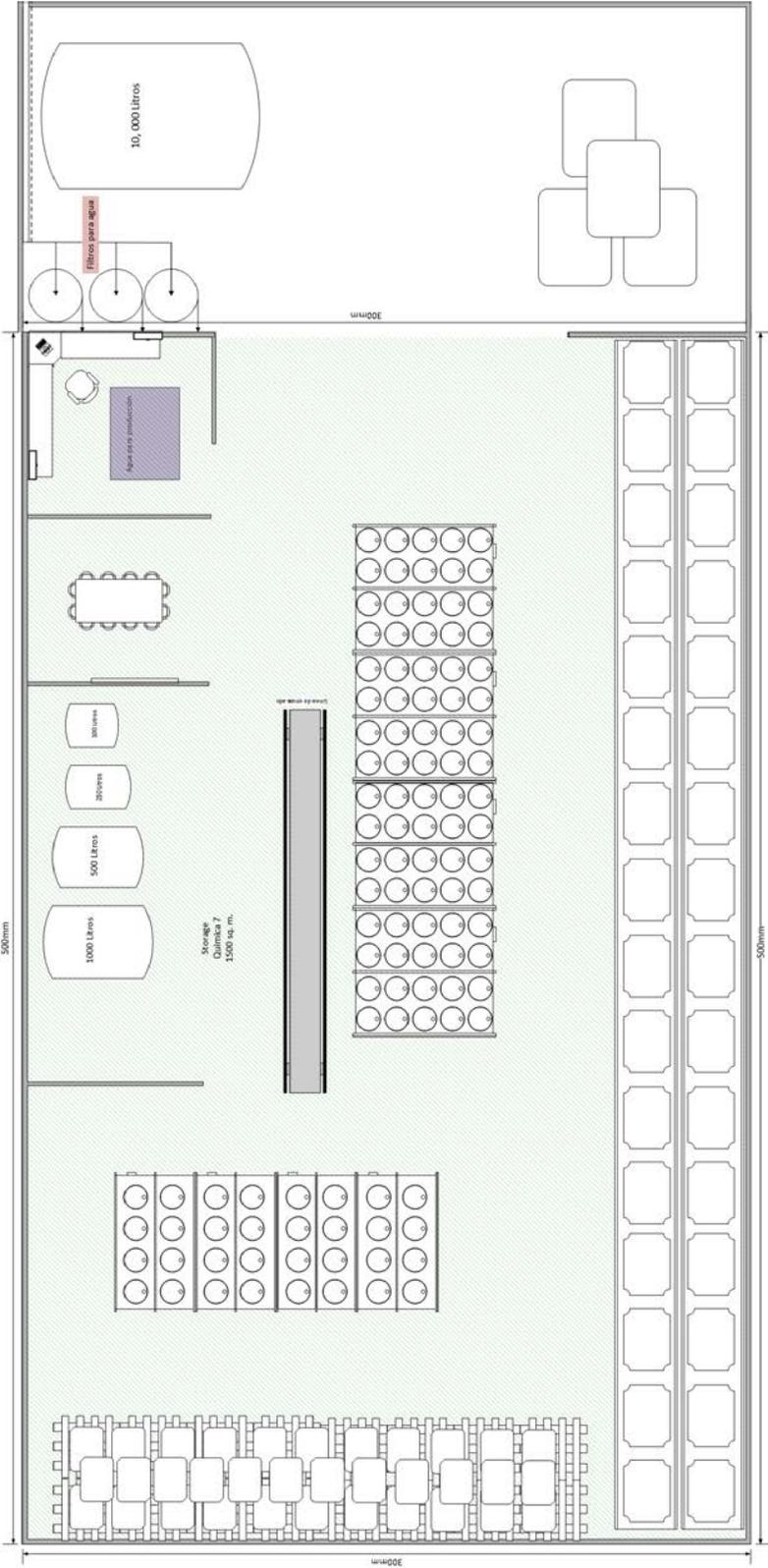
y a los protocolos HTTP y JSON.

- ~ Para el desarrollo web se utiliza el estándar HTML5, CSS y JS, así como guías de usabilidad y diseño de sitios web.

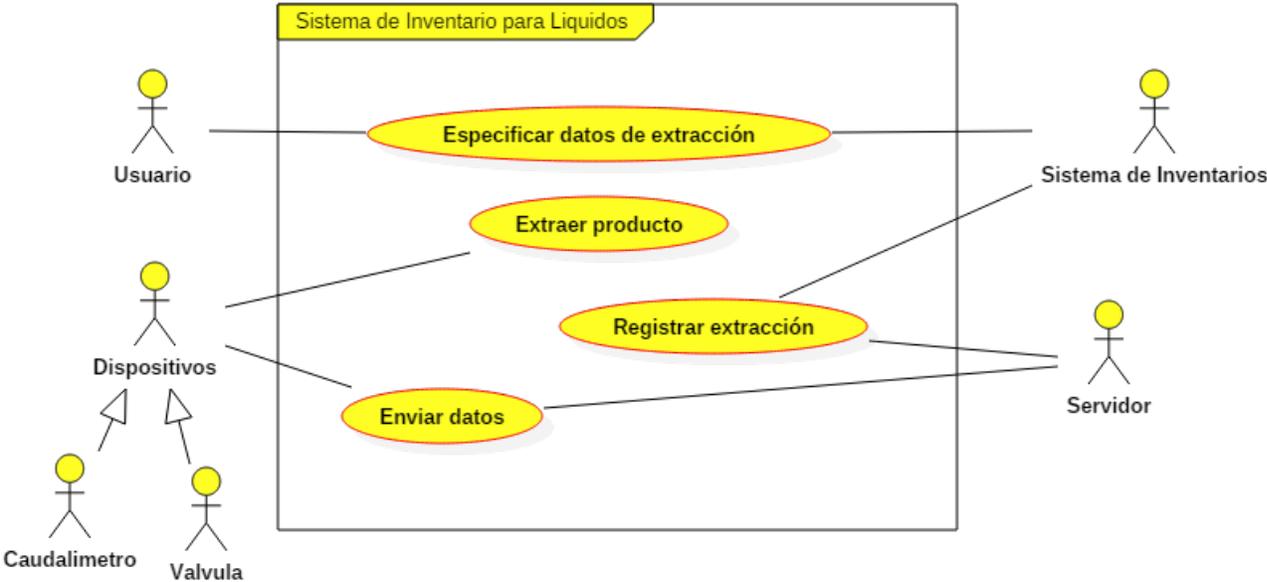
## **Documentación del Sistema**

La documentación del producto abarca meramente los documentos y evidencia que respalden el diseño y modelado del sistema.

# PC2 – Diagrama de Distribución del Almacén Principal



PC 2 – Diagrama de Casos de Uso



## PC2 – Especificación de casos de uso

### Sistema de Inventario para Líquidos Especificación de casos de uso

#### Caso de Uso: Especificar datos de extracción

##### Breve Descripción

El usuario accederá a la interfaz web del microprocesador y especificará los datos del líquido a extraer.

##### Descripción de Actores

###### *Usuario*

Se trata de una persona perteneciente al área de producción, el cual necesita extraer químicos para cumplir con los pedidos recibidos.

###### *Sistema de Inventarios*

Consiste en un sistema de terceros encargado de la gestión de los artículos dentro de los almacenes de la organización.

###### *Microprocesador*

Consiste en una placa de circuitería que concentra la conexión entre dispositivos compatibles, incluye un pequeño procesador que le permite ejecutar un distro Linux. Este OS permite realizar funciones complejas, como ejecutar un web server propio.

##### Precondiciones

~ El sistema debe poder comunicarse con el sistema de inventarios para obtener el catálogo de productos disponibles.

##### Flujo de Eventos Básico

1. El caso de uso comienza cuando un usuario accede a la interfaz web.
2. El sistema se comunica con el sistema de inventarios para obtener el catálogo de productos, clientes y las existencias actuales

3. El sistema desplegará una página donde solicitará que se seleccione un producto, un cliente y se especifique el volumen en mililitros que se extraerán del contenedor.
4. El usuario especifica el producto a extraer, el volumen y el cliente a quien se le atiende con esa extracción.
5. El sistema valida que la cantidad a extraer sea menor o igual al disponible según el sistema de inventarios.
6. El sistema termina la sesión del usuario.
7. Se incluye una referencia al caso de uso "Extraer Producto"
8. El caso de uso termina finaliza.

### **Flujos Alternativos**

#### *Sitio web inaccesible*

Si durante el punto 1 del flujo principal, existiera algún problema con el sitio web, entonces:

1. El sistema mostrara una pantalla de error, informando que en estos momentos se realiza una tarea de mantenimiento.
2. El caso de uso continúa en el paso 6.

#### *Volumen a extraer especificado mayor al disponible en almacén.*

Si durante el punto 5 del flujo principal, la validación dentro de la página web falla, entonces:

1. El sistema indicara al usuario que la cantidad especificada no es válida y debe de corregirse.
2. El usuario especifica un nuevo valor.
3. El caso de uso continúa en el paso 5.

### **Post-Condiciones**

#### *Apertura de Válvula.*

Al término del caso de uso se puede escuchar y observar que la válvula se abre, permitiendo extraer el volumen indicado hacia otro contenedor.

## Caso de Uso: Extraer producto

### Breve Descripción

El sistema coordinara las llamadas entre micro-procesador y microcontrolador para extraer el volumen de líquido especificado.

### Descripción de Actores

#### *Microprocesador*

Consiste en una placa de circuitería que concentra la conexión entre dispositivos compatibles, incluye un pequeño procesador que le permite ejecutar un distro Linux. Este OS permite realizar funciones complejas, como ejecutar un web server propio.

#### *Microcontrolador*

Consiste en una placa de circuitería que concentra la conexión entre dispositivos compatibles, controlada por un microcontrolador re-grabable que permite guardar instrucciones sobre el comportamiento y tratamiento de sus diferentes componentes.

### Precondiciones

~ El sistema conozca el producto y volumen a extraer, así como el cliente a quien se atenderá.

### Flujo de Eventos Básico

1. El caso de uso comienza cuando el micro-procesador le comunica al microcontrolador que un cliente ha pedido la extracción de un producto.
2. El sistema se comunica con el microcontrolador y le indica que es momento de abrir la válvula de apertura.
3. El microcontrolador valida cada vez que detecta flujo dentro del caudalímetro y añade el volumen detectado a un total interno.
4. El sistema valida que el total detectado sea menor al valor especificado por el usuario.
5. El sistema cierra la válvula al detectar que el total de volumen extraído es igual al especificado por el usuario.
6. Incluye referencia al caso de uso "Enviar Datos".
7. El caso de uso termina finaliza.

**Post-Condiciones**

*Registro en sistema de inventarios.*

Al término del caso de uso se puede encontrar una nueva requisición de producto dentro del sistema de inventarios, el cual representa una extracción completada de producto.

**Requerimientos Especiales**

*No aplica*

## Caso de Uso: Enviar Datos

### Breve Descripción

El sistema enviara los datos especificados a través de Internet a un servidor interno, donde también se localiza el sistema de inventario, en donde estos datos serán procesados como una requisición de producto valida.

### Descripción de Actores

#### *Servidor*

Se refiere a una infraestructura localizada dentro de la red interna, la cual se compone de un servidor configurado con las utilidades necesarias para obtener las lecturas enviadas, para guardarlas en una base de datos asegurando su persistencia, que cuenta con las rutinas necesarias para procesar dichos datos y ofrecer información de calidad para el usuario final.

#### *Microcontrolador*

Consiste en una placa de circuitería que concentra la conexión entre dispositivos compatibles, controlada por un microcontrolador re-grabable que permite guardar instrucciones sobre el comportamiento y tratamiento de sus diferentes componentes.

### Precondiciones

- ~ Las lecturas deben estar almacenadas en una estructura de datos dentro del microcontrolador.
- ~ El sistema conoce la dirección IP que tiene el servidor.

### Flujo de Eventos Básico

1. El caso de uso comienza cuando la válvula de paso es cerrada.
2. El sistema almacena los datos de la extracción dentro de una estructura temporal.
3. El sistema traduce la estructura de datos a una con formato JSON.
4. El sistema indica al microcontrolador que genere una conexión a través de Internet a la dirección del servidor.
5. El servidor responde que la conexión fue exitosa.
6. El sistema indica al microcontrolador que envíe la estructura JSON al servidor, a través de la conexión establecida previamente.

7. El servidor responde que la recepción del envío fue correcta y termina la conexión.
8. Se incluye una referencia al caso de uso “Registrar Extracción”.
9. El caso de uso termina.

### **Flujos Alternativos**

#### *Fallo en creación de conexión para Internet*

Si durante el punto 4 del flujo principal, el microcontrolador devuelve un error en la generación de la conexión, entonces:

1. El sistema reintentara la conexión con el servidor.
2. El caso de uso continúa en el paso 5.

#### *Fallo con él envió de la estructura JSON sobre Internet*

Si durante el punto 6 del flujo principal, el microcontrolador devuelve un error en él envió de la estructura JSON, entonces:

1. El sistema reintentara él envió de la estructura al servidor.
2. El caso de uso continúa en el paso 7.

### **Post-Condiciones**

#### *Registro en bitácoras de control*

Al término del caso de uso se puede encontrar un registro con los datos correspondientes al prototipo en la bitácora de conexiones en el servidor.

### **Requerimientos Especiales**

*No aplica*

## **Caso de Uso: Registrar extracción**

### **Breve Descripción**

El sistema llamara a un procedimiento almacenado que le permita realizar en forma de transacción una serie de interacciones con la base de datos que desembocan en un registro exitoso dentro del sistema de inventarios

### **Descripción de Actores**

#### *Servidor*

Se refiere a una infraestructura localizada dentro de la red interna, la cual se compone de un servidor configurado con las utilidades necesarias para obtener las lecturas enviadas, para guardarlas en una base de datos asegurando su persistencia, que cuenta con las rutinas necesarias para procesar dichos datos y ofrecer información de calidad para el usuario final.

#### *Sistema de Inventarios*

Consiste en un sistema de terceros encargado de la gestión de los artículos dentro de los almacenes de la organización.

### **Precondiciones**

- ~ El sistema debe estar activo en el servidor, es decir, el administrador de aplicaciones debe estar ejecutándose.
- ~ El servidor debe permitir que se acceda a dicha aplicación, teniendo habilitada un acceso público.

### **Flujo de Eventos Básico**

1. El caso de uso comienza cuando el servidor recibe una nueva petición con datos
2. El sistema obtiene de la estructura JSON recibida los datos de extracción y ejecuta un procedimiento almacenado dando como parámetros de entrada los datos recibidos.
3. El sistema obtiene del catálogo de clientes el registro completo del cliente seleccionado
4. Se registra un movimiento de extracción dentro de la bitácora de movimientos.
5. Se realizan las restas necesarias y se inserta la extracción del volumen del producto seleccionado.

6. Se inserta el encabezado de una factura de venta, que contiene datos generales del movimiento.
7. Con el folio del encabezado del paso anterior, se inserta el detalle de la factura, la cual consiste en los detalles del producto vendido, su precio con impuestos y su total.
8. Se actualiza el movimiento en la bitácora con el folio del documento generado.
9. Se actualizan las existencias dentro del catálogo de productos disponibles.
10. Se actualiza el folio más reciente al aumentar el uno el registro actual.
11. El caso de uso termina finaliza.

### **Flujos Alternativos**

*No aplica.*

### **Post-Condiciones**

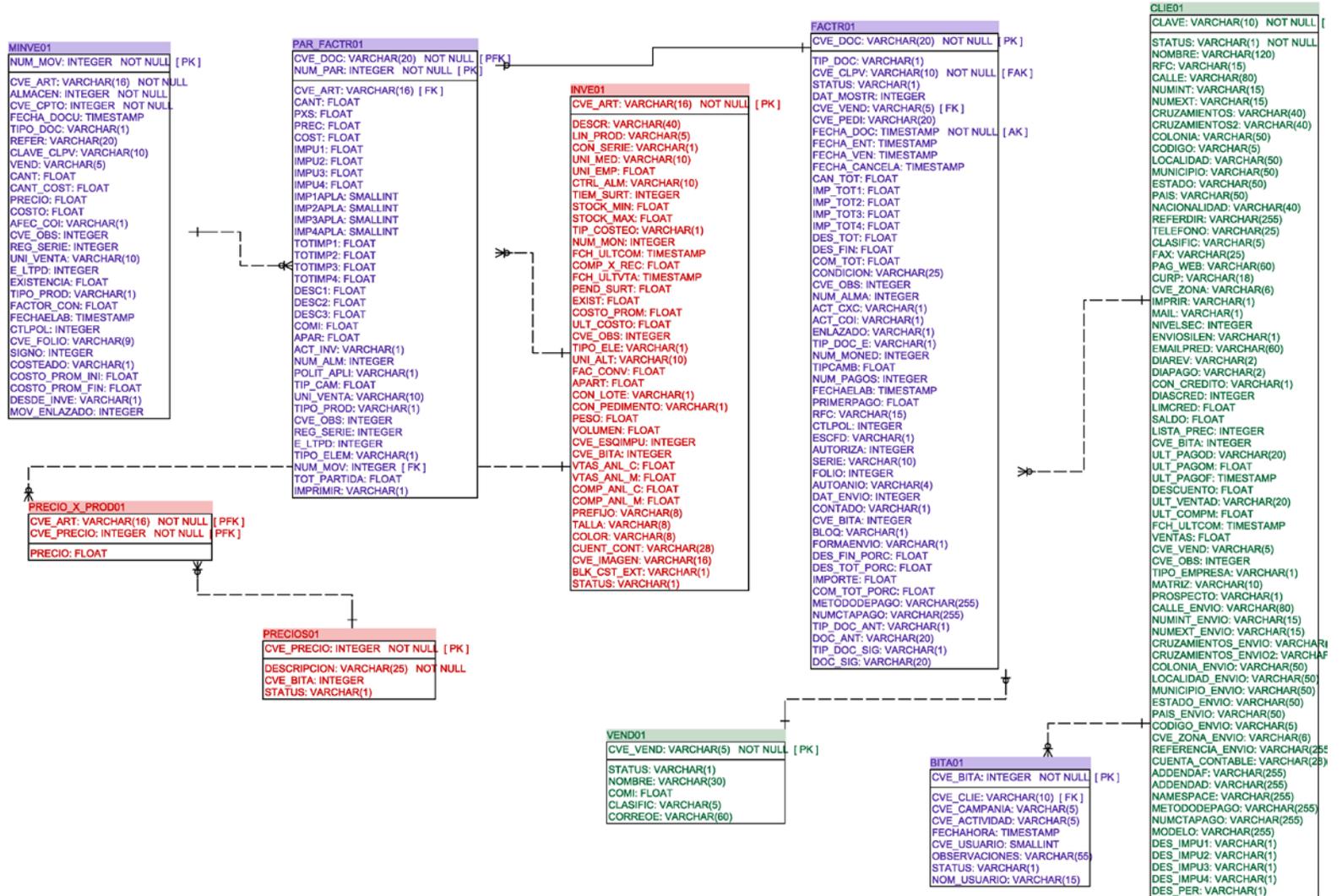
*Registro en sistema de inventarios*

Al término del caso de uso se puede encontrar un nuevo registro de requisición dentro del sistema de inventarios, el cual puede ser consultado e incluso generar reportes o documentos imprimibles de dicho movimiento.

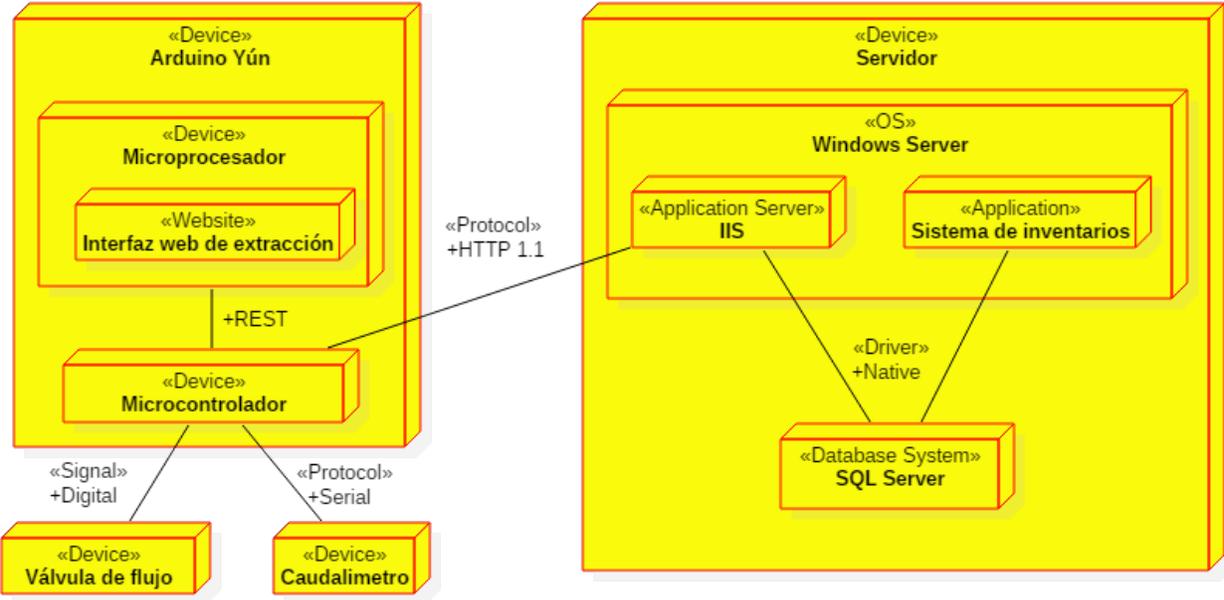
### **Requerimientos Especiales**

*No aplica.*

## PC2 – Diagrama Entidad Relación



# PC2 – Diagrama de Componentes



## Bibliografía

- Adrian McEwen, H. C. (2013). *Designing the Internet of Things*. Wiley John.
- Aguilar, L. J. (2006). *Programacion en C++: Algoritmos, Estructuras de Datos y Objetos*. McGraw Hill.
- Arshdeep Bahga, V. M. (2014). *Internet of Things: A Hands-On Approach*. Vpt.
- Badenes, R. O. (2012). *Sistemas Integrados de Gestión*. Universitat Politècnica de València.
- Blum, J. (2013). *Arduino: Tools And Techniques For Engineering Wizardry*. John Wiley.
- Ch., O. A. (1970). *Introduccion a la Estadistica Descriptiva*. México: UNAM.
- Doukas, C. (2012). *Building Internet of Things with the Arduino*. Createspace.
- Gary Hamel, C. K. (1994). *Competing for the future*. Cambridge, Massachusetts: Harvard Business Review Press.
- Jonás Montilva C., J. B. (2007). *Desarrollo de Software Empresarial*. Universidad de los Andes.
- Kenneth C. Laudon, J. P. (2012). *Sistemas de Información Gerencial*. Pearson.
- N., P. (2015, April 28). *The Internet of Things on a Windows-Based Gateway*. Retrieved from Intel Developer Zone: <https://software.intel.com/en-us/articles/the-internet-of-things-on-a-windows-based-gateway>
- Object Management Group. (2015, Marzo). *OMG Unified Modeling Language Version 2.5*.
- Perez, J. C. (2001). *Sistemas Operativos, Una Vision Aplicada*. McGraw Hill.
- Pressman, R. S. (2002). *Ingenieria de Software, Un Enfoque Practico*. McGraw Hill.
- Russ Miles, K. H. (2006). *Learning UML 2.0*. O'Reilly Media.
- Tannenbaum, A. S. (2009). *Sistemas Operativos Modernos*. Pearson.
- The Eclipse Foundation. (2016, 10). Retrieved from OpenUp Official Documentation: <http://epf.eclipse.org/wikis/openup/>