

NIOTE

Emulador para el desarrollo de proyectos IoT Y analítica de datos



**UNIVERSIDAD DE
SAN BUENAVENTURA
COLOMBIA**

Integrantes:

Camilo Andres Diaz Gomez - 30000050164

Jhonatan Mauricio Villarreal Corredor - 30000051809

Juan Esteban Contreras Diaz – 30000048764

Línea de profundización:

Gestión de Datos

Presentado a:

Ingeniero Andrés Armando Sánchez Martín

Universidad De San Buenaventura

Facultad de Ingeniería

Programa de Ingeniería de Sistemas

Proyecto integrador I

Bogotá D.C.

2020 – 1

Contenido

Resumen.....	7
Introducción.....	8
Justificación	9
Oportunidades	10
Marco conceptual	12
Internet of Things (IoT)	12
Áreas de aplicación.....	12
Modelos de referencia	13
Protocolos	13
Dispositivos	14
Plataformas IoT	15
Análítica de datos	15
Áreas de aplicación de analítica	15
Cadena de valor de los datos	15
Datos	16
Ciclo de la analítica KDD – Metodologías	17
Tipos de analítica	17
Técnicas de analítica.....	18
Tipos de algoritmos.....	18
Simulación	19
Tipos de simulación	19
Fases de estudio de simulación	19
Modelos de simulación	20
Procesos estocásticos	21
Variables aleatorias.....	22
Modelos probabilísticos	22
Números pseudoaleatorios.....	23
Generación de números de pseudoaleatorios	23
Emulación	25
Tipos de emuladores	25
Utilidades comunes de los emuladores.....	26
Marco.....	27
IoT	27
Análítica de datos.....	27

Simulación:	27
Emulación:.....	27
Mapa de Co-Relación de Conocimientos.....	28
Estado del Arte	29
Desarrollo de simulación IoT	31
IOT based wireless sensor network for traffic	31
An IoT simulator in NS3 and a key-based authentication architecture for IoT devices using Blockchain.....	31
A Versatile Emulator of MitM for the identification of vulnerabilities of IoT devices, a case of vulnerabilities of IoT devices, a case of study: smartphones.....	31
Emulation of IoT Devices:	32
OMNeT ++	32
NS3	32
TOSSIM:	32
COOJA:	33
YAFS:	33
Pregunta y Objetivos.....	33
Pregunta Generadora:.....	33
Objetivo General:	34
Objetivos Específicos	34
Alcance y Limitaciones.....	35
Alcances:.....	35
Limitaciones:.....	35
Metodología	36
Fase 1: Análisis	36
Actividades:	36
Entregables:	36
Fase 2: Diseño.....	36
Actividades:	36
Entregables:	36
Fase 3: Construcción	37
Actividades:	37
Entregables:	37
Fase 4: Prueba.....	37
Actividades:	37

Entregables:	37
Cronograma.....	37
Gráfico de apoyo	39
Desarrollo.....	40
Análisis y Diseño Conceptual.....	40
Definición de procesos	40
Definición de requerimientos	40
Trazabilidad.....	49
Diagrama de clases:	49
.....	49
Diseño de detalle.....	50
Arquitectura	50
Muckups:	50
Proceso de construcción y puesta en marcha.....	55
Entorno de desarrollo	55
Prueba – Validación	56
Análisis de Resultados	58
Definición del Caso de Estudio	58
Puesta en marcha del Caso de Estudio	58
Análisis estadístico/probabilístico	59
Conclusiones.....	62
Conclusiones	62
Lecciones aprendidas y experiencia	62
Trabajos futuros	62
Bibliografía y Referencias.....	63

Tabla de ilustraciones

Ilustración 1 Modelo general de la arquitectura de una red IoT	13
Ilustración 2 Mapa de Co-Relación de Conocimientos	28
Ilustración 3 Diagrama de apoyo (Metodología)	39
Ilustración 4 Diagrama de clases	49
Ilustración 5 Arquitectura del proyecto	50
Ilustración 6 Pantalla Principal.....	50
Ilustración 7 Panel de proyectos	51
Ilustración 8 Panel de emulación.....	51
Ilustración 9 Panel OutPut del proyecto	52
Ilustración 10 Panel de nodos IoT	52
Ilustración 11 Navegación MenuBar.....	53
Ilustración 12 Navegación MenuBar (Editar)	53
Ilustración 13 Navegación MenuBar (Ejecutar)	54
Ilustración 14 Navegación MenuBar (Ayuda).....	54
Ilustración 15 Navegación MenuBar (Ventana)	55
Ilustración 16 Entorno de desarrollo	56
Ilustración 17 Panel Principal funcionando	57
Ilustración 18 Respuestas con respecto a la visibilidad del estado del sistema y la óptima aplicación de los iconos seleccionados	60
Ilustración 19 Respuestas con respecto a la estética y diseño minimalista y el gusto por los encuestados a la interfaz	60
Ilustración 20 Respuestas con respecto a la estética y diseño minimalista y la óptima distribución de los paneles en la interfaz principal.....	60
Ilustración 21 Respuestas con respecto a la estética y diseño minimalista y la buena selección de colores para el tema oscuro en la interfaz principal	61

Tabla de tablas

Tabla 1 Etapas para poder realizar una cadena de valor	Error! Bookmark not defined.
Tabla 2 Comparación de proyectos	Error! Bookmark not defined.
Tabla 3 Permitir crear una red	Error! Bookmark not defined.
Tabla 4 RF2 Permitir cargar configuraciones previas de red .	Error! Bookmark not defined.
Tabla 5 RF3 Permitir borrar cualquier red	Error! Bookmark not defined.
Tabla 6 RF4 Permitir reiniciar cualquier red.....	Error! Bookmark not defined.
Tabla 7 RF5 Permitir configurar la red.....	Error! Bookmark not defined.
Tabla 8 RF6 Permitir controlar la red.....	Error! Bookmark not defined.
Tabla 9 RF7 Permitir configurar puertas de enlace.....	Error! Bookmark not defined.
Tabla 10 RF8 Permitir configurar nodos IoT	Error! Bookmark not defined.
Tabla 11 RF9 Permitir configurar sensores	Error! Bookmark not defined.
Tabla 12 RF10 Permitir configurar actuadores.....	Error! Bookmark not defined.
Tabla 13 RF11 Permitir guardar la red	Error! Bookmark not defined.
Tabla 14 RF12 Permitir la creación de nuevos proyectos	Error! Bookmark not defined.
Tabla 15 RF13 Permitir abrir los proyectos ya creados.....	Error! Bookmark not defined.
Tabla 16 RF14 Permitir seleccionar la ubicación de guardado	Error! Bookmark not defined.

Tabla 17 RF15 Permitir crear una carpeta de raíz.....	Error! Bookmark not defined.
Tabla 18 RFN16 Generación de archivos planos.....	Error! Bookmark not defined.
Tabla 19 RFN17 Interfaz amigable con el usuario.....	Error! Bookmark not defined.
Tabla 20 Casos de usos - Requerimientos.....	Error! Bookmark not defined.

Resumen

Describa brevemente la problemática, a quienes afecta, como se pretende solucionar e impactos que genera con la realización del proyecto en la práctica de ingeniería de sistemas II. Mínimo 250 palabras, máximo 600.

Introducción

Párrafo informando el contenido total del documento, y como fue desarrollado. La introducción es una guía para el lector en donde se presenta, además de una motivación y la puesta en valor de la contribución del proyecto, la estructura general que encontrará en el documento. Mínimo 500 palabras.

Justificación

En la actualidad, una gran parte de los objetos de uso diario por las personas están conectados a internet por diferentes propósitos, este concepto es conocido como IoT "Internet of Things", el cual, *"se refiere a la interconexión en red de objetos cotidianos, que a menudo están equipados con inteligencia ubicua."* (Xia, Yang, Wang, & Vinel, 2012).

Gracias a esto, la conexiones IoT se han enlazado a la vida cotidiana por el avance de las tecnologías en la sociedad que han evolucionado a una velocidad en la que muchas personas, empresas, negocios, entre otros, están adquiriendo estas nuevas tecnologías que les ayudan a mejorar el rendimiento en diferentes aspectos por que los dispositivos IoT están encargados de la obtención de datos y su envío a la nube, permitiendo la conexión e intercambio de información entre estos objetos. Según el centro de investigación SAP, estos objetos están perfectamente integrados a la red de información, lo que hace que se pueda interactuar con los mismos a través de internet, pudiendo consultar o editar su estado a tiempo real (Abasolo, Carrera, Gordillo, & Romero, 2013). Ver de donde salió este.

Actualmente, hay demasiadas personas que se desempeñan en el diseño, instalación y mantenimiento de estas conexiones, estos a su vez buscan alguna ayuda para poder desarrollar sus pruebas sin la necesidad de gastar muchos recursos; por esta razón, las dificultades más recurrentes para la realización de estas conexiones y su posterior análisis, es su alto costo económico, la cual, limita un desarrollo de estas para que sean optimas, eficientes y sobre todo que sea poco costo, y también, el tiempo en la construcción de estas conexiones que es una de las razones por la que estas conexiones pueden costar más o menos dependiendo del propósito y presupuestos.

Oportunidades

En el mercado encontramos algunos simuladores, entre ellos tenemos (Iglesia, 2019):

- **NS-3(Network Simulator versión 3):** Simulador de redes de sensores que trabaja en diferentes escalas, aplicaciones de campo y condiciones de campo (Riley & Henderson, 2010).
- **OMNeT ++:** Es una biblioteca de simulación de c++, orientado a la simulación de redes de comunicación y sistemas distribuidos pero usado también para simular una red de sensores IoT (Varga, 2010).
- **TOSSIM:** Simulador para redes de sensores inalámbricos TinyOS (Levis & Lee, 2003).
- **Avrora:** Simulador de red de sensores de nivel de instrucción y con precisión de ciclo (Titzer, 2005).
- **Cooja:** Cooja es un emulador construido sobre el sistema operativo Contiki. Contiki OS es un pequeño sistema operativo que está optimizado para usarse con muy poca huella de memoria y un procesamiento mínimo de capacidades. Está especialmente diseñado para ser utilizado para Internet de las cosas. Cooja emula diferentes propiedades funcionales de dispositivos IoT como la temperatura de emulación. Trabajo relacionado a detección, detección de presión, etc. También puede emular la comunicación entre pares cercanos entre los nodos y la transmisión de mensajes cercanos en función de la distancia. Eso emula todas las funcionalidades al construir sobre la capa de abstracción del sistema operativo contiki. Como emula sobre un sistema operativo real, puede hacer perfiles basados en tiempo de diferentes funciones (Dunkles, y otros, 2011).
- **OPNET:** OPNET es el simulador de red comercial que es mantenido por las tecnologías de Riverbed y solo admite la plataforma Windows (Riverbed Technologies, 2017).
- **NetSim NetSim:** Es otro simulador y emulador comercial que se usa para probar IoT redes y aplicaciones (Tetcos, 2017).
- **NÚCLEO CORE:** Es el emulador de investigación abierta común que se utiliza para emular PC y redes en una o varias PCs. Utiliza la pila de red del núcleo de FreeBSD para emulación. Sin embargo, no es compatible con muchos de los protocolos y pilas de red utilizados en la comunicación IoT (Ahrenholz, Danilov, Henderson, & Kim, 2008).

Según la información de cada uno de los emuladores, se evidencia que todos simulan solamente redes de sensores, algunas incluso solo un tipo de sensor como es el caso de TOOSSIM (Levis & Lee, 2003). A diferencia de estos NIOTE permite recrear una red de dispositivos IoT más completa con distintos tipos de sensores y actuadores. (Sánchez, Barreto, Ochoa, Villanueva, 2019), (Gan, 2017)

Marco conceptual

En este apartado se explicará todos los conceptos a tener en cuenta para el entendimiento, desarrollo y presentación del proyecto.

Internet of Things (IoT)

El Internet de las cosas es un nuevo paradigma del mundo moderno el cual es la conexión de varios nodos IoT (cosas) que contienen integrados sensores, actuadores, gateways, plataformas (software) y otros componentes; para generar conexiones e intercambio de información por medio del internet para posteriormente ser procesada con un objetivo. En la actualidad, este concepto es muy ubicuo (cotidiano) en el día a día, aunque no sea muy percibido, ya que la mayoría de las cosas de uso diario como lo son los electrodomésticos, celulares, entre otras cosas; lo integran. (Xia, Yang, Wang, & Vinel, 2012), (CambioDigital, 2018)

En la actualidad, como fue mencionado anteriormente la mayoría de los dispositivos integran este nuevo concepto y esto por su gran versatilidad a los diferentes usos que se le puedan aplicar a cada uno y no es extraño que la humanidad se está anticipando a un cambio tecnológico en donde IoT será tan importante e indispensable para la vida cotidiana de los humanos, algo no tan diferente a lo que estamos viviendo ahora.

Áreas de aplicación

La humanidad está pasando por una transformación en el cual en un futuro dependerá de la tecnología directamente, por esta razón, IoT va a ser un gran pilar para este cambio, por esta razón, está siendo implementado en diferentes sectores como lo son agricultura, salud, transporte y logística, seguridad, entre otros.

Beneficios

La aplicación de este concepto es variada y funcional, ya que, es versátil para el objetivo que se quiera cumplir con estas conexiones por que al ser generados datos por el ambiente donde se encuentre montado y por último ser analizados dependiendo el objetivo cuya ejecución haya sido planeada (Universidad de Alcalá, 2019).

Modelos de referencia

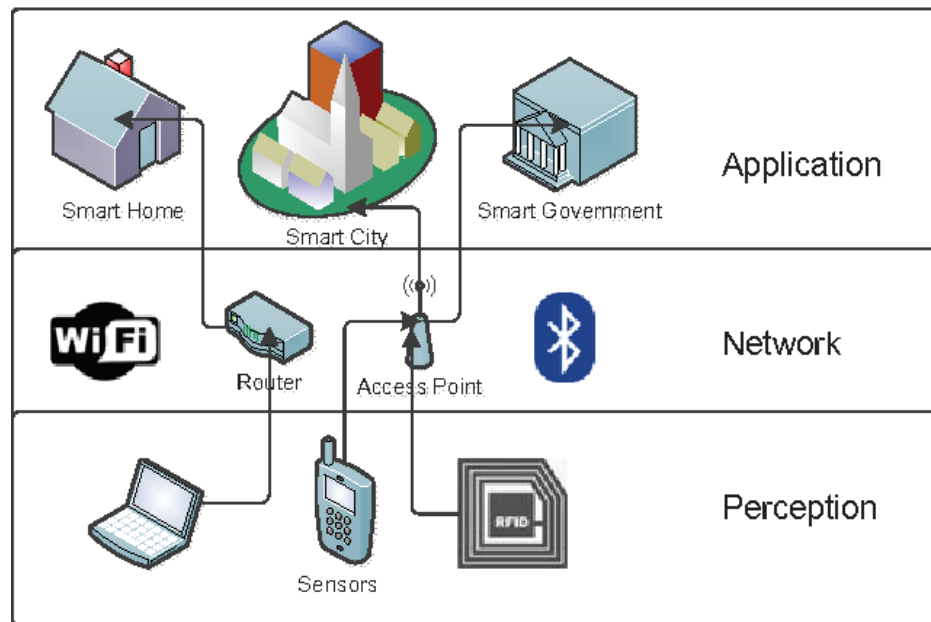


ILUSTRACIÓN 1 MODELO GENERAL DE LA ARQUITECTURA DE UNA RED IoT

En la anterior Ilustración 1 Modelo general de la arquitectura de una red IoT se puede evidenciar un modelo general de IoT, donde se puede ver dividido en 3 grandes campos, hay que resaltar que los nombres de cada parte del modelo pueden variar, pero su definición y función serán igual. En primer lugar, podemos ver percepción, el cual contiene todos los sensores que recolectan la información para ser enviada a la red o nube por medio de los gateways y actuadores que son los que reciben toda la información recogida por los sensores a través de la red y gracias a esto actuar dependiendo el requerimiento o el objetivo al que se quiera llegar; como se mencionó anteriormente toda la información es enviada a la red o nube, la cual almacena toda esta información para ser procesada y con su posterior análisis de los datos, estar lista para la aplicación en las "cosas" (Crespo Moreno, 2018).

Protocolos

Los protocolos son una guía para saber cómo realizar una acción, de esta manera, los protocolos en IoT son los métodos en la que dos o más componentes se comunican por medio de la red, de esta manera, garantizar la consistencia de la información.

Comunicación

Según la empresa Opiron Electronics el protocolo por comunicación es *“método para comunicar datos entre máquinas, ni más ni menos. Este método, en realidad queda definido tanto por el medio físico como por el medio informático en la que se intercambian los datos.”* De esta manera, en este protocolo se pueden encontrar (Semle, 2016), (Opiron Electronics, 2017).

Transporte

Este protocolo está dirigido para conectar software mediante una dirección especificada, y en el nivel de enlace solo abra un enlace, el otro software siempre estará abierto para la conexión, esto gracias a que el host estará abierto (NEO.LCC, s.f.).

Operación

Este protocolo está dirigido para conectar entre dos o más entidades para la comunicación y transmisión de información entre estas mediante los protocolos definidos (AprendiendoArduino, 2018).

Dispositivos

Como se definió anteriormente, IoT no puede funcionar sin los nodos IoT lo cuales son los diferentes dispositivos, en su mayoría físicos, los cuales garantizan el tratamiento de datos rápido, seguros, y eficientes.

Sensores

Los sensores son los dispositivos que recogen la información gracias al ambiente en donde se encuentra funcionando, de esta manera, por medio de los gateways es enviada todos los datos para luego ser procesada y analizada en la red.

Actuadores

Los actuadores son los dispositivos que recogen la información producida por los sensores y analizada en la red, todo esto por medio de los gateways, para posteriormente por medio de la aplicación de control funcionar dependiendo la situación que en el que esté funcionando.

Gateway

Los gateways son dispositivos intermediarios entre los sensores o actuadores y la red. Generalmente son físicos o software los cuales son los que reciben la información gracias a los sensores para posteriormente enviarla a la red y después de ser procesada recibirla y enviarlo a los actuadores. (Crespo Moreno, 2018).

Plataformas IoT

En la actualidad, en cuanto a plataformas para el monitoreo de estas conexiones existe una gran variedad, de cierto modo, IoT se esta convirtiendo en un pilar para la sociedad y por esta razón han desarrollado plataformas nuevas y cada día mejores para el buen uso, desarrollo, gestión y mantenimiento de estas. TEERM (Quiñones Cuenca, González Jaramillo, Torres, & Miguel , 2017)

Analítica de datos

La analítica de datos es la utilización de información que se puede tener u obtener de manera digital, con el propósito de extraer la mejor información para poder tomar las mejores decisiones (Gibbs, 2012).

Áreas de aplicación de analítica

Las áreas donde se puede aplicar la analítica son un poco extensas ya que muchas actividades o procedimientos que realizamos es necesario hacer una investigación anteriormente para poder obtener unos resultados apropiados para poder analizar y así poder utilizarlos para realizar (Joyanes Aguilar, 29 de mayo del 2019); muchas de las áreas pueden ser de economía, probabilidad, administración, web, inteligencia artificial etc (Gomez-Aguilar, Garcia-Peñalvo, & Theron, 2014).

Cadena de valor de los datos

La cadena de valor de datos como su propio nombre indica, son una gran cantidad de datos que representan cierta información (Quintero, 2006); en el cual ciertas empresas ya están destinadas a prestar estos servicios; la cadena tiene varias etapas para su realización. Esto también lleva a una extensa investigación mediante el cual se utilizan ciertas etapas para poder realizar (Hergert & Morris, 1989)

TABLA 1 ETAPAS PARA PODER REALIZAR UNA CADENA DE VALOR

Collection	Publication	Uptake	impact
Identify	Analyze	Connect	Use
Collect	Release	Incentivize	Change
Process	Disseminate	Influence	reuse

La cadena de valor proporciona cierto modelo de aplicación el cual permite representar todas las actividades de cualquiera empresa y también proporciona un procedimiento para el desarrollo de ventajas (Hergert & Morris, 1989).

Datos

Representación simbólica de alguna información o procedimiento de manera en la cual puede ser almacenado y analizado para poder realizar ciertas operaciones para poder generar información adecuada; toda Información debe estar disponible en todo momento y un precio justo en el cual los datos deben tener ciertos privilegios para poder distribuirlos, reutilizarlos. (Loukides, 2011)

Datos abiertos

Los datos abiertos son datos o información que se pueden utilizar, publicar y reutilizar tantas veces quiera sin ninguna repercusión, pero los datos no pueden ser modificados (Hernandez Perez, 2013).

Fuentes de datos

La fuente de datos son conjuntos de información con sus respectivos datos recolectados para su respectivo análisis en la cual son fuentes de información para nivel informático y analítico (Diaz, Escriba, & Murgui, 2002).

Estructurados

Son la mayoría de los datos que se pueden encontrar almacenados en una base de datos; la cual se muestran en fila y columnas, tienen definido su longitud el formato en el cual se encuentra el tamaño que tiene (Hernandez & Rodriguez, 2008).

No estructurados

Son esencialmente datos binarios que no tiene una estructura u organización que no tiene algún valor al utilizarlos hasta que son organizados y almacenados (Hassan, Domingo-Ferrer, & Soria-comas, 2018).

Semi - estructurados

Son datos que no son organizados en un repositorio, pero tiene información importante como metadatos (datos que están cerca de los datos) la cual hace que se pueda procesar más fácilmente los datos (Raposo, 2007).

Ciclo de la analítica KDD – Metodologías

El proceso de extraer conocimiento a partir de grandes volúmenes de datos la cual es un proceso automático en el que se combinan descubrimiento y análisis. La cual tiene unos pasos a seguir (Nigro, Corti, & Terren, 2004).

- Selección.
- Preprocesamiento/limpieza.
- Transformación/reducción.
- Minería de datos.
- Interpretación/evaluación

Tipos de analítica

Es necesario saberlos ya que con el aumento de la cantidad de datos que generan actualmente empresas, negocios etc.

“un conjunto de métodos de análisis matemático y estadístico que sirve para identificar patrones de comportamiento, pronósticos, escenarios “que pasaría si”, entre otros” (Davenport y Harrys, 2017).

Descriptiva

¿qué sucedió? un resumen del desempeño del total de las actividades empresariales (cuatro tipos de analítica de retail que todo comercio necesita en 2018, 2018).

Diagnostica

¿Por qué está pasando? Tiene en cuenta los antecedentes de lo que se quiere analiza para dar un informe más acertado con sus respectivas herramientas para poder eliminar

el problema (cuatro tipos de analítica de retail que todo comercio necesita en 2018, 2018).

Predictiva

¿qué podría pasar? Tienen como objetivo identificar la probabilidad que ocurra algo en el futuro que no perjudique o perjudique a análisis realizado (cuatro tipos de analítica de retail que todo comercio necesita en 2018, 2018).

Prescriptiva

¿qué deberíamos hacer?, entendimiento de lo que ha sucedido, por qué ha sucedido y un procedimiento en el cual podría suceder con el paso del tiempo (cuatro tipos de analítica de retail que todo comercio necesita en 2018, 2018).

Técnicas de analítica

pronosticar resultados potenciales sobre la base de posibles variaciones en las variables, muchas veces la base o la raíz que tiene la analítica es entender lo que ha pasado o lo que está pasando en el momento; todo esto para poder adquirir el conocimiento para mejorar las decisiones hacia el futuro, muchas veces se utilizan técnicas de estadística y matemáticas para poder lograr el objetivo (Garcia, 2006).

MACHINE LEARNING

La técnica de machine learning es un apoyo para el conocimiento de ciertas generaciones, tiene una organización en el cual es el auto aprendizaje que muestra estadísticas con una gran velocidad de respuesta. Esto es una disciplina la cual trata de crear o construir modelos complejos, también algoritmos que buscan llegar o alcanzar una predicción, el cual funciona sin dar alguna instrucción o orden la cual busca que se actualice automáticamente con los datos que adquirió para poder adaptarse a la situación que se les presente (Snoke & HalanLarochelle, 2012).

Tipos de algoritmos

Algoritmo es una secuencia de instrucciones secuenciales, gracias al cual pueden llevarse a cabo ciertos procesos y darse respuesta a determinadas necesidades o decisiones (Corso & Lorena, 2009).

Supervisados

"los algoritmos trabajan con datos "etiquetados" (labeled data), intentado encontrar una función que, dadas las variables de entrada (input data)" (Corso & Lorena, 2009).

No supervisados

"no se dispone de datos "etiquetados" para el entrenamiento. Sólo conocemos los datos de entrada, pero no existen datos de salida que correspondan a un determinado input" (Corso & Lorena, 2009).

Simulación

La simulación es una representación exacta del comportamiento interno de un evento o fenómeno, diseñado para obtener el mismo resultado, características, información entre otros, consumiendo menos recursos de los que consumiría ejecutar el modelo real, sin necesidad de realizar dicho evento para obtener un análisis o estudio del resultado con una menor inversión (Castellanos Hernández & Chacon Osorio, 2006).

Tipos de simulación

Simulación de situaciones: Permite simular una situación física o real y observar su comportamiento (Castellanos Hernández & Chacon Osorio, 2006).

Simulación de realizar alguna situación: Son aquellos que permiten experimentar una situación como si el usuario u sujeto estuviera en ella, un simulador de vuelo es un ejemplo de esto, permite al usuario pilotear un avión sin estar en uno realmente (Castellanos Hernández & Chacon Osorio, 2006).

Fases de estudio de simulación

- Definición de objetivos: Se deben establecer los objetivos que se pretenden conseguir con la simulación, los efectos que causara y las respuestas a responder con este estudio (García Sánchez & Ortega Mier, 2006).
- Definición del sistema: Definir los elementos que harán parte del sistema teniendo en cuenta el sistema a emular (García Sánchez & Ortega Mier, 2006).
- Elaboración del modelo conceptual: A partir de los objetivos planteados anteriormente se crea un modelo conceptual, el cual debe ser sencillo (solo enfocarse en lo necesario para simular) y específicamente diseñado para

cumplir dichos objetivos. El modelo conceptual debe representar sencillez y a su vez representar el realismo del sistema a emular (García Sánchez & Ortega Mier, 2006).

Este modelo conceptual debe ser evaluado y comprobar que refleje fielmente el sistema que se desea emular teniendo en cuenta los objetivos que debe cumplir (Coss Bu, 2003).

- **Elaboración del sistema comunicativo:** Los diseñadores del modelo conceptual son distintos muchas veces a los programadores del simulador. Para su comunicación entre si debe ser eficaz, por esta razón los diagramas de flujo son una opción útil para representar los eventos en el simulador como lo son los datos, el proceso, una decisión un avance en la simulación etc. (García Sánchez & Ortega Mier, 2006).
- **Construcción y verificación de modelo informático:** Una vez verificado el modelo conceptual se escoge un lenguaje apto para para la programación del simulador, este lenguaje debe permitir la correcta emulación como fue planeada, se debe escoger el más conveniente (García Sánchez & Ortega Mier, 2006).
- **Validación final:** Una vez construido el modelo de simulación creado anteriormente, es necesario hacer pruebas para verificar su correcto funcionamiento, en las cuales los resultados deberán ser similares a los esperados y si es posible se comparará con los resultados del sistema real al cual se está simulando (Coss Bu, 2003).

Modelos de simulación

Hay diversos modelos de simulación los cuales serán mencionados a continuación, pero nos centraremos más en el modelo Estocástico:

- **Estático:** La simulación no depende del tiempo (García Sánchez & Ortega Mier, 2006).
- **Dinámico:** La simulación depende del tiempo, sus procesos pueden variar respecto al tiempo que va transcurriendo (García Sánchez & Ortega Mier, 2006).

- **Determinístico:** Su ejecución será siempre igual, el valor de su resultado será ya esperado, no tiene ninguna variable o proceso al azar (García Sánchez & Ortega Mier, 2006).
- **Estocástico:** Contiene variables o procesos al azar el cual puede variar el resultado de la simulación (García Sánchez & Ortega Mier, 2006).
- **Discreto:** Varía dependiendo de sucesos que ocurran en la simulación del modelo (García Sánchez & Ortega Mier, 2006).
- **Continuo:** Tiene un rango de tiempo el cual es previamente establecido. Sin importar los sucesos que ocurran seguirá ejecutándose.
- **Físicos:** Se basan en eventos físicos o fenómenos que ocurren y no son posibles de controlar y estudiar (García Sánchez & Ortega Mier, 2006).

Procesos estocásticos

Los procesos estocásticos es una colección de variables aleatorias infinitas que se basan en el cambio o evolución de una variable con respecto al tiempo o en función de otra variable como lo puede ser la temperatura, el cambio climático entre otras (Myriam Muñoz de ózak, 1991).

Las variables aleatorias que dependen del tiempo son aquellos fenómenos que evoluciona al azar a lo largo del tiempo, el tiempo tomará diferentes valores en dicho conjunto donde la colección de variables aleatorias se verá afectadas por esto tomando diferentes valores según T (Myriam Muñoz de ózak, 1991).

El conjunto de variables que dependen de otra variable son aquellos que su evolución no dependen del tiempo sino de otro fenómeno (Myriam Muñoz de ózak, 1991), estas variables pueden tomar valores directamente proporcionales a este fenómeno no perteneciente al conjunto dicho anteriormente es decir que los valores de las variables aumentaran si el valor del fenómeno aumenta y disminuirá si este disminuye, por otro lado las variables aleatorias que son indirectamente proporcionales al fenómeno harán todo lo contrario si el valor del fenómeno aumenta, el valor de las variables disminuirá y si este disminuye las variables aumentaran.

Variables aleatorias

Las variables aleatorias son parte fundamental de una simulación, ya que los sistemas requieren diferentes tipos de datos no siempre serán los mismos para ejecutar un evento simulado, por esta razón es de vital importancia crear variables aleatorias ya que necesitamos que la simulación sea lo más apegado posible a la realidad (Eduardo García Dunna, 2013).

El modelo que queremos construir debe estar compuesto de variables aleatorias que interactúen entre sí, para asemejarlo a la realidad (Eduardo García Dunna, 2013). Una variable aleatoria es una representación de un suceso o un número de una parte del evento que estamos intentando emular, es decir que un suceso en el sistema puede variar en ese mismo proceso u otro proceso independiente, por ejemplo, si queremos emular un sensor de temperatura, una de las variables aleatorias será la temperatura ya que esta puede cambiar con el paso del tiempo o ser diferente en otra prueba del simulador (Eduardo García Dunna, 2013).

Existen dos tipos de variables aleatorias, la primera es la variable aleatoria discreta, la cual se caracteriza por ser de solo números enteros es decir no puede tomar valores como 10,97 u otro tipo de número que no es un entero (Eduardo García Dunna, 2013), por otra parte la variable aleatoria continua es más caracterizada por su uso para las mediciones en este caso si puede contener valores decimales, retomando el ejemplo del medidor de temperatura podemos decir que esta simulación consta de variable aleatoria continua ya que la temperatura es una medida y puede tener parte decimal (Eduardo García Dunna, 2013).

Modelos probabilísticos

Cuando hablamos de un modelo probabilístico nos referimos a un conjunto de datos obtenidos por diversas repeticiones de un evento aleatorio usados para poder predecir el comportamiento de este evento con los mismos o diferentes datos para las futuras repeticiones de dicho evento (Leónardo Darío Bello Parias, 2000), esta serie de repeticiones permiten asemejar el modelo que se está construyendo con datos aleatorios a una conjunto de datos de una población mayor, con esto nos referimos a

la simulación más acercada posible de un evento real mediante la prueba y repetición del modelo que se está simulando.

Existen varios modelos probabilísticos para variables aleatorias:

- Distribución Uniforme
- Distribución Gamma
- Distribución Exponencial
- Distribución Ji-dos
- Distribución Normal
- Distribución t Student
- Distribución F de Sendecos
- Distribución normal bivalente

Los modelos probabilísticos son basados en hipótesis y se compone por ecuaciones las cuales relacionan las diversas variables aleatorias (Carlos Gamero Burón, 2015), estos modelos son la representación más viable de una hipótesis para un evento que este compuesto de variables aleatorias por lo cual debe ser rectificado correctamente y probado una y otra vez.

Números pseudoaleatorios

Una simulación, muchas veces se compone de variables aleatorias es decir números al azar, para conseguir esto los números pseudoaleatorios son parte fundamental en este proceso de simulación, su nombre está compuesto de dos palabras, "Pseudo" lo cual significa falso y "aleatorio", se le denomina falso debido a que es imposible generar números completamente aleatorios, al no ser posible generar números completamente aleatorios los números pseudoaleatorios son creados a partir de algoritmia determinística con parámetros de arranque, esto nos permitirá generar números que se comportaran similarmente a números totalmente aleatorios es decir números sin correlación entre ellos mismos permitiéndonos simular el comportamiento aleatorio de las variables en el evento que queremos simular (Eduardo García Dunna, 2013).

Generación de números de pseudoaleatorios

Para hacer la generación de los números Pseudoaleatorios debemos tomar un espacio o rango lo suficientemente grande para ello, es decir cuente con demasiados números

en secuencia para una vida útil prolongada (Eduardo García Dunna, 2013). Es necesario este conjunto tan grande porque al hacer una simulación pequeña se necesitarán un conjunto de números mínimo, pero si queremos hacer una aun mayor este número incrementara, pero al hacer la simulación no puede basarse en solo un resultado para ello es necesaria la simulación una y otra vez con números distintos es por esto por lo que es necesario dicho conjunto lo suficientemente grande para satisfacer esta necesidad (Eduardo García Dunna, 2013).

Para aprobar el uso de estos números el conjunto de números pseudoaleatorios se debe someter a ciertas pruebas que nos permitan comprobar la independencia entre ellos y que estos sean uniformes, para ellos se mencionaran unas pruebas estadísticas para la aprobación de este conjunto se debe asegurar que los números de un conjunto deben ser uniformemente distribuidos lo cual significa que en los subintervalos haya la misma cantidad de números del conjunto, deben ser continuos, la media del conjunto debe ser equivalente a $\frac{1}{2}$ y la varianza también debe ser $\frac{1}{2}$ (Eduardo García Dunna, 2013).

Ventajas de la simulación

- La simulación permite ahorrar recursos para obtener los posibles resultados del comportamiento de un evento (García Sánchez & Ortega Mier, 2006).
- A partir de la simulación es posible trabajar mejor los experimentos debido a su mejor manejo en las condiciones de dicho experimento (García Sánchez & Ortega Mier, 2006).
- Es posible a partir de la simulación comparar y escoger el sistema más viable dependiendo de una necesidad (García Sánchez & Ortega Mier, 2006).
- La simulación puede permitir una mejor comprensión del evento que está simulando (García Sánchez & Ortega Mier, 2006).
- Con una simulación es posible hacer diferentes experimentos y su reacción a estos, los cuales no son posibles con el modelo físico el cual se pretende obtener esta nueva información (García Sánchez & Ortega Mier, 2006).

Desventajas de la simulación

- Una vez creada la simulación es posible ahorrar tiempo en la obtención de los datos del modelo simulado, pero para crear la simulación lleva tiempo y estudios

los cuales no son mayores los recursos que requerirá usar el modelo real (García Sánchez & Ortega Mier, 2006).

- La simulación debe ser exacta al modelo real pero aun así se puede generar datos no correctos o no exactos algunas veces a los reales (García Sánchez & Ortega Mier, 2006).

Emulación

La emulación es la imitación a la parte externa de un sistema en la cual el usuario tendrá la misma experiencia como si fuera el sistema original (Roldán Carrasco, 2007), debido a esto la emulación solo permite dar la experiencia de un sistema (cumplir sus funciones) sin serlo.

A diferencia de un simulador, el emulador debe imitar exactamente el comportamiento externo del sistema original de esta manera debe dar la experiencia original al usuario como si este estuviera delante del sistema original (Roldán Carrasco, 2007).

Tipos de emuladores

- Emulador de software: Los emuladores de software son los emuladores más populares debido a su utilidad a la hora de simular sistemas operativos o emular consolas de videojuegos.

Los sistemas operativos son emulados por las máquinas virtuales, estas permiten usar un sistema operativo exactamente como si fuera el original, se pueden usar varias máquinas virtuales en un mismo PC siendo independientes una de la otra al tiempo (Roldán Carrasco, 2007).

La emulación de videojuegos permite ejecutar un juego en una plataforma para la cual no fue creado, este tipo de emulación está centrado en el entretenimiento.

- Emuladores de hardware: Este tipo de emuladores permite comprobar el funcionamiento de un sistema antes de ser fabricado de forma física (Roldán Carrasco, 2007).
- Emuladores interpretadores: Mediante bucles el sistema permite aprender de otra para poder emularla siguiendo los pasos de captura, decodificación y

ejecución, el bucle permite actualizar la información obtenida constantemente (Roldán Carrasco, 2007).

- Emuladores recopiladores: Permite convertir el código máquina de una máquina a un nuevo código para otra máquina en tiempo de ejecución recompilándolo de forma binaria (Roldán Carrasco, 2007).

Utilidades comunes de los emuladores

Los emuladores permiten usar un sistema que un usuario necesita, pero no fue creado para el sistema que está usando (Roldán Carrasco, 2007), se puede usar un emulador y este permitirá usar el sistema que se desea proporcionando la misma experiencia que el sistema original.

Otra utilidad es la posibilidad de usar varios sistemas emulados al tiempo y que estos no sean dependientes entre sí.

"Node-RED es una herramienta de programación para conectar dispositivos de hardware, API y servicios en línea de formas nuevas e interesantes" (Node-Red, 2013). "Proporciona un editor basado en navegador que facilita la conexión de flujos utilizando la amplia gama de nodos en la paleta que se pueden implementar en su tiempo de ejecución con un solo clic" (Node-Red, 2013).

"SimpleIoT Simulator™ es un simulador de dispositivo / sensor IoT fácil de usar que crea rápidamente entornos de prueba formados por miles de sensores y puertas de enlace, todo en una sola computadora" (SimpleSoft, s.f.).

Marco

Para el desarrollo de nuestro proyecto que es la fabricación de un emulador IoT es necesario conocer conceptos básicos, el concepto inicial es el de IoT, porque es importante IoT en nuestro proyecto nuestro proyecto necesita IoT porque vamos a realizar un simulador IoT en el cual necesitamos tener claro cómo se comporta una red IoT para poder simularlo, otro concepto que necesitamos el de data analytics el cual como nuestro proyecto va apoyar el desarrollo de proyectos de data analytics necesitamos saber cómo el data analytic necesita los datos para poder generar los proyectos simulación e emulación, la simulación es un concepto matemático el cual lo que permite hacer es crear modelos matemáticos para poder emular el comportamiento de las cosas el cual en nuestro caso son los dispositivos IoT, el cual nuestro proyecto necesita estos cuatro grandes conceptos para poder desarrollar nuestro proyecto.

IoT

Internet of Things es la interconexión entre dispositivos los cuales están integrados por sensores y actuadores para generar intercambio de información entre estos por medio de la red, todo lo anterior, para ser procesada y realizar acciones teniendo en cuenta la información recogida.

Analítica de datos

La analítica de datos es un desarrollo en el cual se utilizan todo tipo de información en el cual se puede encontrar de forma digital con el fin de poder obtener la mejor información de ellos.

Simulación:

La simulación es un proceso con el cual se replica un evento o un proceso de la manera más acertada posible, con el fin de obtener los mismos resultados del evento, pero con menos recursos gastados y de una manera más fácil de manipular.

Emulación:

La emulación permite a un usuario obtener la misma experiencia y las mismas opciones que ofrece un sistema sin la necesidad de que este sistema sea usado por el usuario, de

esta manera la emulación es una réplica de los usos y la experiencia lo más exacta posible de un sistema.

Mapa de Co-Relación de Conocimientos

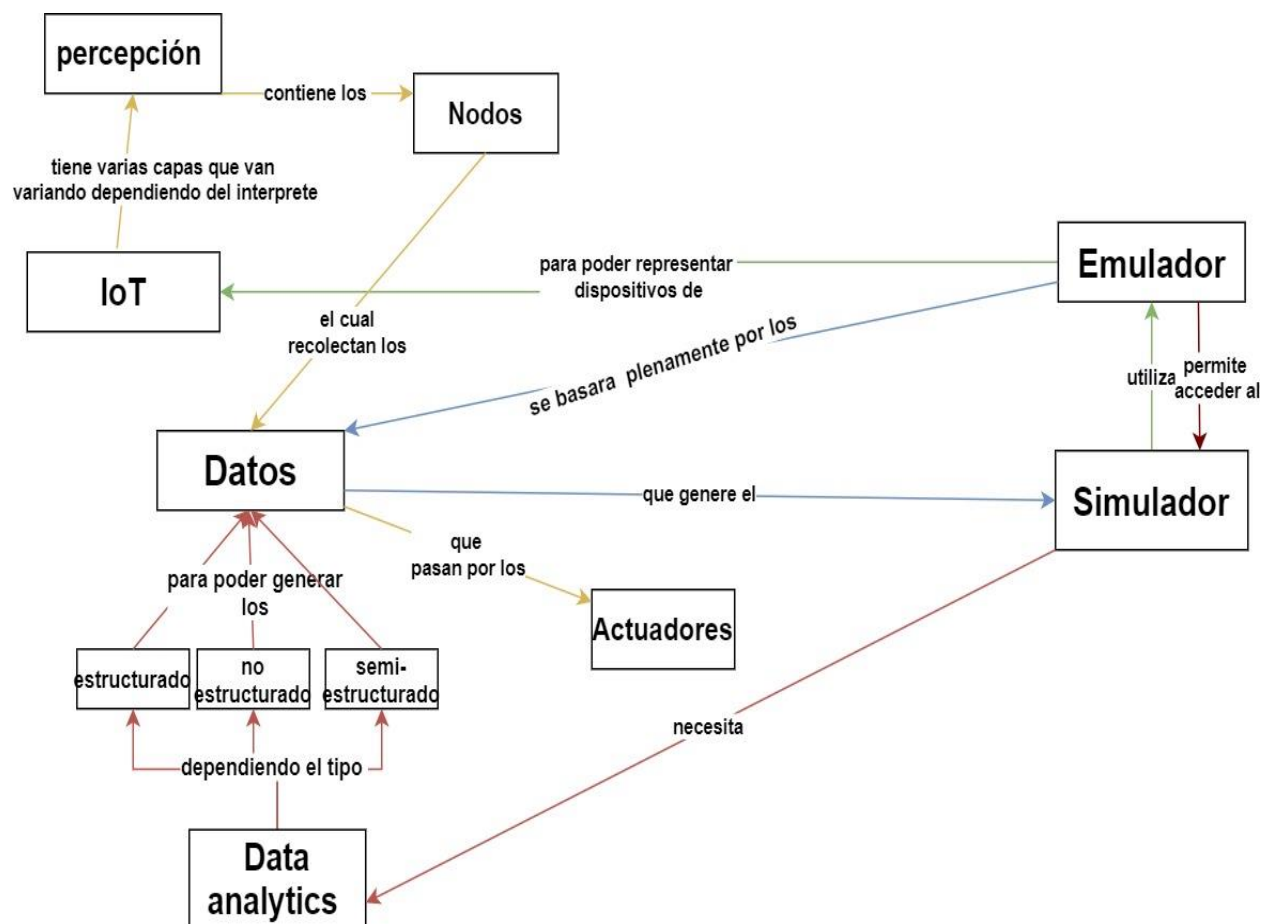


ILUSTRACIÓN 2 MAPA DE CO-RELACIÓN DE CONOCIMIENTOS

En este mapa se va a dar a conocer los conceptos más importantes en el desarrollo de nuestro emulador y la relación que tiene entre ellos el cual tiene una información para su explicación; si muchos de los conceptos tienen mucha interacción entre si hay una mejor información entre ellos.

Estado del Arte

En el estado del arte buscamos respectivos trabajos o programas en los cuales se utilicen los conceptos que queremos agregar a nuestro emulador, la cual nos basamos en diferentes investigaciones y la mayoría o todos los proyectos investigados tiene la parte simulación de sensores y a la hora de investigar en la parte de actuadores no se encontraron muchos de los proyectos son muy escasos los proyectos con este concepto; escogimos proyectos en los que tuvieran conceptos que deseamos utilizar en nuestro emulador.

- Simulación de datos: Uno de los aspectos más importante en nuestro proyecto es la simulación de datos ya que sin la información que estos generan no se podría realizar su respectivo funcionamiento a la hora de realizar una simulación de un sensor o actuador.
- Patrón de modelo MVC: En la parte del patrón que utilizaremos para la realización del emulador fue el patrón MVC ya que nos muestra una mejor organización de todos los paquete y clases que utilizaremos.
- Lenguaje java: para la realización del emulador elegimos el lenguaje java ya que muchos de los emuladores ya existentes no están en este lenguaje programado y la mayor parte están programados mediante el lenguaje C o C++ ; y nos sentimos más cómodos a la hora de programar.
- Conexiones entre sensores y actuadores: para la realización del emulador se escogió este factor diferenciador porque muchos de los emuladores y simuladores de redes IoT no contienen relaciones de este tipo, en algunos solo contienen conexión entre sensores y/o otros con actuadores.
- S.O Windows: se escogió este fator porque en este caso será un límite del proyecto, aun así, fue escogido este por la gran cantidad de usuarios de este sistema operativo.

TABLA 2 COMPARACIÓN DE PROYECTOS

proyectos	Simulación de datos	Patrón de diseño MVC	Lenguaje JAVA	Conexión entre sensores	Conexión entre actuadores	S.O Windows	Utilidad en procesos educativos, investigación y proyectos
Desarrollo de simulación IoT	x	x		X		X	
IOT based wireless sensor network for traffic	x			x		x	X
An IoT simulator in NS3 and a key-based authentication architecture for IoT devices using Blockchain	x						x
A Versatile Emulator of MitM for the identification of vulnerabilities of IoT devices, a case of vulnerabilities of IoT devices, a case of study: smartphones	x					x	X
Emulation of IoT Devices	X						
OMNeT ++	x			x			X
NS3				x		x	x
TOSSIM	X			X			
COOJA	x			X	x		
YAFS				x	X		

Desarrollo de simulación IoT

Desarrollar mediante las arquitecturas basadas en Internet of Things un simulador que sea capaz de registrar conjunto de dispositivos y el cual genere automáticamente datos de entrada para afectar a su estado. El cual mediante un extenso análisis de aplicaciones similares en diferentes campos en el cual tengan en común el concepto IoT, se desarrolló mediante el entorno de patrón de diseño Modelo-Vista-controlador el cual se propone para mejora a futuro el informe y el simulador nos ayuda a visualizar mejor el aspecto al cual le debemos dar a nuestro emulador (Torres Bataller, 2016).

IOT based wireless sensor network for traffic

En el proyecto en el cual se utilizan sensores basados en IoT nos muestra y nos da un punto de apoyo para la realización de nuestro proyecto el cual mostraremos los diferentes tipos de sensores los cuales son los más utilizados para así no tener que agregar muchos sensores el cual haga el proyecto muy lento y no tenga una interfaz amigable con el usuario; nos ayudara a establecer bien los programas para poder sacar lo mejor de casa sensor y actuador el cual vallamos a utilizar (Huang, Wang, Hou, Zhang, & Zhang, 2018).

An IoT simulator in NS3 and a key-based authentication architecture for IoT devices using Blockchain

Este proyecto nos retroalimentara en el nuestro ya que se hace uso de simuladores IoT para la estación de BlockChain sin gastar recursos y tiempo, lo cual, es nuestro objetivo general, por esta razón, la forma en que van a ser usados los diferentes simuladores para este proyectó nos mostrara un punto de vista diferente el cual es de los usuarios finales quienes usaran el emulador. Por lo tanto, el anterior proyecto nos dará una perspectiva diferente a programador y también los requerimientos que solicita un usuario para simular un proyecto con IoT (Gan, 2017).

A Versatile Emulator of MitM for the identification of vulnerabilities of IoT devices, a case of vulnerabilities of IoT devices, a case of study: smartphones

Este proyecto nos servirá para proteger nuestro emulador de errores de seguridad que se puedan generar al momento de montar la red simulada por el usuario, de esta manera, se evitaran problemas legales al ser usado NIOTE para proyectos de estas redes

y luego ser montadas en el mundo físico. (Amine Khelif, Lorandel, Romain, Regnery, & Baheux, 2019).

Emulation of IoT Devices:

Esta tesis nos muestra protocolos como CoAP que permite la conexión entre dispositivos con el internet evidenciando de la misma manera su funcionamiento simulado en el emulador construido en esta, también nos es útil para darnos una guía al trabajo hecho por ellos y la manera en la que lograron emular los dispositivos y los actuadores mediante la representación como objetos IPSO tomando también la forma en la que interactúan dichos dispositivos con el entorno y puedan ser controlados de una manera simple (Mäkinen, 2016).

OMNeT ++

La aplicación OMNeT ++ nos da una perspectiva para el desarrollo de la arquitectura que le queremos dar para la agregación de las herramientas que utilizan las simulaciones, nos da una fuente de información amplia si queremos realizar el emulador en diferentes sistemas operativos en conclusión es uno de los mejores para la realización de sensores (Varga, 2016).

NS3

En este caso, NS3 es un simulador de sensores IoT, el cual está orientado a protocolos de red. Este simulador nos ayudara a comprender el comportamiento como los datos e información que pueden tomar los sensores sobre una red, también, nos ayudara a comprender los protocolos de comunicación MQTT ya que este es compatible con casi todos. (NSNAM, 2011).

TOSSIM:

El simulador nos da un punto de vista al funcionamiento que le deseamos dar al emulador en la parte de la simulación de los sensores y actuadores el cual nos muestra el comportamiento de muchos sensores funcionando al mismo (Levis, Lee, Welsh, & Culler, 2003).

COOJA:

Este proyecto que está basado en IoT nos es útil por la forma de conectar los dispositivos a una red nos permite ver desde otra perspectiva la manera en la que se hará la interconexión con los dispositivos en el simulador, debemos tener en cuenta la red y la IP que usara cada dispositivo que se planea simular a partir de un modelo físico (Mehmood).

YAFS:

Este proyecto al estar enfocado en el almacenamiento de datos lo cual una red IoT debe usar para mandar señales entre dispositivos dependiendo de los datos de estos mismos para interactuar entre ellos, nos permite comprender mejor la forma en la que interactúan los dispositivos y el medio que usan para hacer dicha interacción (Isaac Lera, 2019).

De acuerdo con todo lo anterior todos los proyectos el cual escogimos nos proporcionan mucha información el cual debemos tener en cuenta para la realización de nuestro proyecto. cómo nos lo demuestra la prime segunda columna el cual nos informa que es muy importante que el proyecto tenga la simulación de datos y la conexión entre sensores, tener un diseño MVC (modelo, vista y controlador) nos diferencia de y que muy pocos proyectos tiene la conexión entre actuadores y nos muestra que nuestro proyecto tiene varios conceptos muy importantes como que es pertinente, innovador, factible y viable gracias a los aspectos que elegimos.

Pregunta y Objetivos

Pregunta Generadora:

¿Cómo reducir tiempos en el diseño y desarrollo de proyectos IoT y analítica de datos implementando modelos de simulación para reducir los costos económicos en la fase de implementación de los proyectos?

Objetivo General:

Desarrollar un emulador de dispositivos IoT que pueda hacer el diseño lógico de la red y genere datos para facilitar el diseño y prueba conceptual de proyectos de IoT o analítica de datos.

Objetivos Específicos

- Especificar los requerimientos de software para emular dispositivos, protocolos y arquitecturas de redes IoT.
- Diseñar los modelos de simulación necesarios para la emulación de dispositivos y enlaces de comunicación IoT.
- Construir el emulador IoT implementando los modelos de simulación diseñados sobre una aplicación con interfaz gráfica amigable.
- Realizar pruebas funcionales para comprobar el óptimo funcionamiento del emulador.

Alcance y Limitaciones

Alcances:

- El emulador se compromete a tener una interfaz amigable con el usuario con el fin de un fácil y entendimiento ante la aplicación.
- El emulador se compromete a guardar los datos generados en archivos planos.
- El emulador se basará en un modelo realístico para la generación de datos aleatorios.

Limitaciones:

El emulador no:

- funcionará en ordenadores con sistemas operativos diferentes a Windows.
- tendrá todos los sensores y actuadores que hay disponibles en la actualidad.
- guardara los datos en una base de datos.
- va a generar los datos simulados con precisa precisión, aun así, se buscará la forma óptima y eficiente para realizarlos.
- generará estadísticas de los datos generados en su funcionamiento.
- tendrá por el momento un manual para el entendimiento de las diferentes funcionalidades que tendrá este.
- generará datos de datos aleatorios para la simulación de la red.
- tendrá en funcionamiento el BackEnd, en esta instancia, el emulador funciona parcialmente el FrontEnd.

Metodología

Fase 1: Análisis

En esta etapa se realizará el análisis del emulador, con lo cual, análisis de protocolos, análisis de las arquitecturas y se realizará los diferentes diagramas UML.

Actividades:

- Actividad 1: Análisis de requerimientos.
- Actividad 2: Análisis de arquitectura y protocolos.
- Actividad 3: Diagrama Casos de uso.
- Actividad 4: Diagrama de clases.
- Actividad 5: Diagrama de paquetes.
- Actividad 6: Diagrama de arquitectura.

Entregables:

- ANEXO: Documentación de los requerimientos, arquitectura y protocolos escogidos, y diagramas.

Fase 2: Diseño

En esta etapa se realizará el diseño del emulador, con lo cual, se realizará un análisis para encontrar un diseño sencillo, puro e intuitivo con el usuario.

Actividades:

- Actividad 7: Análisis del diseño.
- Actividad 8: Diseño del MockUp.
- Actividad 9: Búsqueda de iconos.
- Actividad 10: Diseño y modificación de la interfaz en el IDE con base al MockUp.
- Actividad 11: Socialización y aprobación del diseño.

Entregables:

- ANEXO: MockUp.

Fase 3: Construcción

En esta etapa se realizará el emulador con ayuda del IDE Netbeans 8.2, por lo tanto, se harán procesos de implementación del MockUp y realización de la lógica en el interior de este.

Actividades:

- Actividad 12: Conectividad de ventanas y botones
- Actividad 13: Conexión de lógica con interfaz

Entregables:

- ANEXO: Emulador con interfaz gráfica funcionando en gran parte.

Fase 4: Prueba

En esta etapa se realizarán las diferentes pruebas a el emulador, están con base a modelos realizados de conexiones IoT, con ello se buscara encontrar errores en el emulador en su fase de código, simulación y emulación.

Actividades:

- Actividad 14: Pruebas a código base del emulador, con este se busca revisar que no haya ningún error lógicamente en interfaz y lógica de programación en el emulador.
- Actividad 15: Pruebas del emulador, con este se busca probar la simulación de los datos, comportamiento de los diferentes nodos IoT, entre otras.

Entregables:

- ANEXO: Documento con pruebas realizadas.

Cronograma

El cronograma es uno de los elementos que está integrado en la planeación de proyectos, está compuesto por actividades donde se mostrara el tiempo estimado y/o específico en el que se va a realizar dicha acción.

Gráfico de apoyo

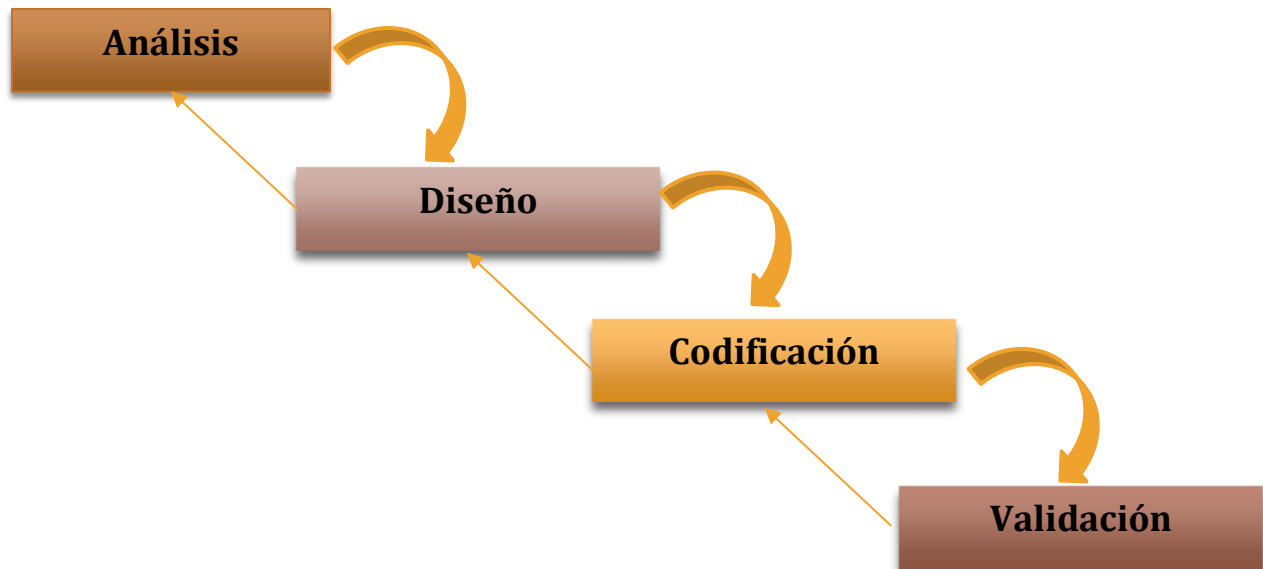


ILUSTRACIÓN 3 DIAGRAMA DE APOYO (METODOLOGIA)

Desarrollo

En el desarrollo de este proyecto como inició, se establecieron unos requerimientos y objetivos los cuales se cumplirán a medida que se llevara a cabo este, de esta manera, hasta la fecha nuestro proyecto ha avanzado de la siguiente forma mediante algunas imágenes.

Análisis y Diseño Conceptual

En esta sección se presenta lo que se realizó para la comprensión del problema, la definición de procesos, la definición de requerimientos, casos de uso o el modelo conceptual de los productos que comprenden la solución presentada en el proyecto.

Definición de procesos

La definición de procesos que realizamos en el desarrollo de nuestra aplicación o emulador fueron los siguientes:

Cada integrante del grupo creo o utilizo una cuenta de GitHub.

Se creó un brazo para cada miembro del grupo para poder subir avances del proyecto y el master los actualizaba para poder descargarlos.

Realizamos un grupo en el cual podíamos dar ideas de como queríamos del boceto del emulador.

Implementación de los bocetos mediante la creación del código del emulador.

Cada integrante presenta avances al grupo y se decide si se implementa, se mejora o se niega la idea.

Definición de requerimientos

ID	RF1	Nombre	Permitir crear una red		
Tipo (necesario/ deseable)			Autor del requerimiento		Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras		15 de mayo de 2020
¿Crítico?		Actores		Prioridad de desarrollo	
Si		Usuario		Alta	
Entrada				Salida	
El usuario puede crear una red en el sistema.				El sistema creara y guardara la red creada por el usuario.	
Descripción					
Descripción		Permitir al usuario crear una red para el uso del emulador.			

TABLA 3 PERMITIR CREAR UNA RED

ID	RF2	Nombre	Permitir cargar configuraciones previas de red	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El usuario crea una configuración de red.			El sistema la guardara para su reutilización.	
Descripción				
Descripción		El usuario puede reutilizar la configuración que uso por última vez dado que el sistema guardara esta configuración.		

TABLA 4 RF2 PERMITIR CARGAR CONFIGURACIONES PREVIAS DE RED

ID	RF3	Nombre	Permitir borrar cualquier red	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores			Prioridad de desarrollo
Si	Usuario			Alta
Entrada				Salida
El usuario elije una red para eliminarla.				El sistema eliminara la red seleccionada.
Descripción				
Descripción		El usuario puede elegir la red que desea eliminar y el sistema borrara la red.		

TABLA 5 RF3 PERMITIR BORRAR CUALQUIER RED

ID	RF4	Nombre	Permitir reiniciar cualquier red	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El usuario elije una red para reiniciarla.			El sistema reiniciara la red seleccionada.	
Descripción				
Descripción		El usuario puede elegir la red que desea reiniciar y el sistema reiniciara la red.		

TABLA 6 RF4 PERMITIR REINICIAR CUALQUIER RED

ID	RF5	Nombre	Permitir configurar la red	
Tipo (necesario/ deseable)			Autor requerimiento del	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores			Prioridad de desarrollo
Si	Usuario			Alta
Entrada				Salida
El usuario elije una red para configurarla.				El sistema cambiara la configuración de acuerdo a lo solicitado por el usuario de la red seleccionada.
Descripción				
Descripción		El usuario puede elegir la red que desea configurar y los aspectos que desea cambiar y el sistema brindara la interfaz y opciones esenciales para lograr la configuración deseada.		

TABLA 7 RF5 PERMITIR CONFIGURAR LA RED

ID	RF6	Nombre	Permitir controlar la red	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El usuario elije una red que desea controlar.			El sistema permitirá hacer uso de la red elegidas.	
Descripción				
Descripción		El usuario puede elegir la red que desea controlar y el sistema brindara una interfaz que permita acceder al uso de la red elegida.		

TABLA 8 RF6 PERMITIR CONTROLAR LA RED

ID	RF7	Nombre	Permitir configurar puertas de enlace	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El usuario elije las puertas de enlace que desea configurar.			El sistema permitirá cambiar la configuración de las puertas de enlace.	
Descripción				
Descripción		El usuario puede elegir las puertas de enlace que desea configurar y el sistema permite hacer cambios y guardarlos de la configuración realizada por el usuario.		

TABLA 9 RF7 PERMITIR CONFIGURAR PUERTAS DE ENLACE

ID	RF8	Nombre	Permitir configurar nodos	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Diaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El usuario elije los nodos que desea configurar.			El sistema permitirá al usuario configurar los nodos deseados.	
Descripción				
Descripción	El usuario puede elegir los nodos que desea configurar, el sistema brindara la interfaz para la configuración y guardara la información de configuración de los nodos.			

TABLA 10 RF8 PERMITIR CONFIGURAR NODOS IOT

ID	RF9	Nombre	Permitir configurar sensores	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El usuario elije los sensores que desea configurar.			El sistema permitirá al usuario configurar los sensores como el dese y los quiere utilizar.	
Descripción				
Descripción		El usuario puede elegir los sensores que desea configurar, el sistema brindara la interfaz para la configuración y guardara la información de configuración de los nodos.		

TABLA 11 RF9 PERMITIR CONFIGURAR SENSORES

ID	RF10	Nombre	Permitir configurar actuadores	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El usuario elije los actuadores que desea configurar.			El sistema permitirá al usuario configurar los actuadores deseados.	
Descripción				
Descripción		El usuario puede elegir los nodos que desea configurar, el sistema brindara la interfaz para la configuración y guardara la información de configuración de los nodos.		

TABLA 12 RF10 PERMITIR CONFIGURAR ACTUADORES

ID	RF11	Nombre	Permitir guardar la red	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El usuario podrá tener la opción de guardar			El sistema permitirá al usuario guardar el proyecto que esté realizando.	
Descripción				
Descripción		El usuario puede elegir en emulador la opción de guardar todos los proyectos que esté realizando sin tener que perder el proceso que lleva.		

TABLA 13 RF11 PERMITIR GUARDAR LA RED

ID	RF12	Nombre	Permitir la creación de nuevos proyectos	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	de
Si	Usuario		Alta	
Entrada			Salida	
El usuario podrá realizar la creación de todos los proyectos relacionados con redes que desee			El sistema permitirá al usuario tener la creación de sus proyectos.	
Descripción				
Descripción		El usuario puede elegir en la interfaz la opción de crear todos los proyectos que desee, teniendo en cuenta su respectivo proyecto abierto.		

TABLA 14 RF12 PERMITIR LA CREACIÓN DE NUEVOS PROYECTOS

ID	RF13	Nombre	Permitir abrir los proyectos ya creados	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	de
Si	Usuario		Alta	
Entrada			Salida	
El usuario podrá abrir los proyectos que tengan la extensión. niote .			El sistema permitirá al usuario mediante la interfaz la opción de abrir proyectos.	
Descripción				
Descripción		El usuario puede elegir mediante la interfaz para la opción de abrir cualquier proyecto que tenga con una extensión. niote ; el cual no podrá abrir sin esa extensión.		

TABLA 15 RF13 PERMITIR ABRIR LOS PROYECTOS YA CREADOS

ID	RF14	Nombre	Permitir seleccionar la ubicación de guardado		
Tipo (necesario/ deseable)			Autor del requerimiento		Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras		15 de mayo de 2020
¿Crítico?		Actores		Prioridad de desarrollo	
Si		Usuario		Alta	
Entrada				Salida	
El usuario elije la ubicación la cual desea guardar el proyecto.				El sistema permitirá al usuario guardar todo proyecto que esté realizando.	
Descripción					
Descripción		El usuario puede elegir la ubicación en el cual desea guardar el archivo que está utilizando.			

TABLA 16 RF14 PERMITIR SELECCIONAR LA UBICACIÓN DE GUARDADO

ID	RF15	Nombre	Permitir crear una carpeta de raíz	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores			Prioridad de desarrollo
Si	Usuario			Alta
Entrada				Salida
El sistema automáticamente creara una carpeta de raíz				El sistema le creara al usuario una carpeta en su respectivo ordenador.
Descripción				
Descripción		El sistema automáticamente creara una carpeta de raíz para los trabajos que se desarrollen el emulador, si el usuario no selecciona alguna ubicación para el archivo.		

TABLA 17 RF15 PERMITIR CREAR UNA CARPETA DE RAÍZ

ID	RFN16	Nombre	El emulador tendrá archivos planos	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El emulador trabajara con archivos planos			El emulador guardara los datos en el formato de archivos planos	
Descripción				
Descripción		El emulador no tendrá una base de datos que acompañe, si no estarán en un formato de archivos planos que tendrán la información recolectada.		

TABLA 18 RFN16 GENERACIÓN DE ARCHIVOS PLANOS

ID	RFN17	Nombre	El sistema tendrá una interfaz amigable con el usuario	
Tipo (necesario/ deseable)			Autor del requerimiento	Fecha
Necesario			Camilo Andrés Díaz, Jonatan Mauricio Villarreal, Juan Esteban Contreras	15 de mayo de 2020
¿Crítico?	Actores		Prioridad de desarrollo	
Si	Usuario		Alta	
Entrada			Salida	
El sistema tendrá interfaz fácil			El sistema mostrara lo más sencillo posible una interfaz para que el usuario.	
Descripción				
Descripción	El sistema mostrar al usuario una interfaz fácil de utilizar para la realización de proyectos IoT.			

TABLA 19 RFN17 INTERFAZ AMIGABLE CON EL USUARIO

Trazabilidad

TABLE 4 TRAZABILIDAD

Caso de Uso	Descripción CU	Requerimientos
CU01	CRUD : crear, registro , modificación, eliminación de una red	R01, R03,R12
CU02	cargar configuraciones	R02, R05,
CU03	reiniciar redes	R04
CU04	controlar las redes	R06
CU05	configurar todo para la red	R07,R08,R09,R10
CU06	guardar redes	R11
CU07	Abrir nuevos proyectos	R13
CU08	seleccionar ubicación, guardado	R14

TABLA 20 CASOS DE USOS - REQUERIMIENTOS

Diagrama de clases:

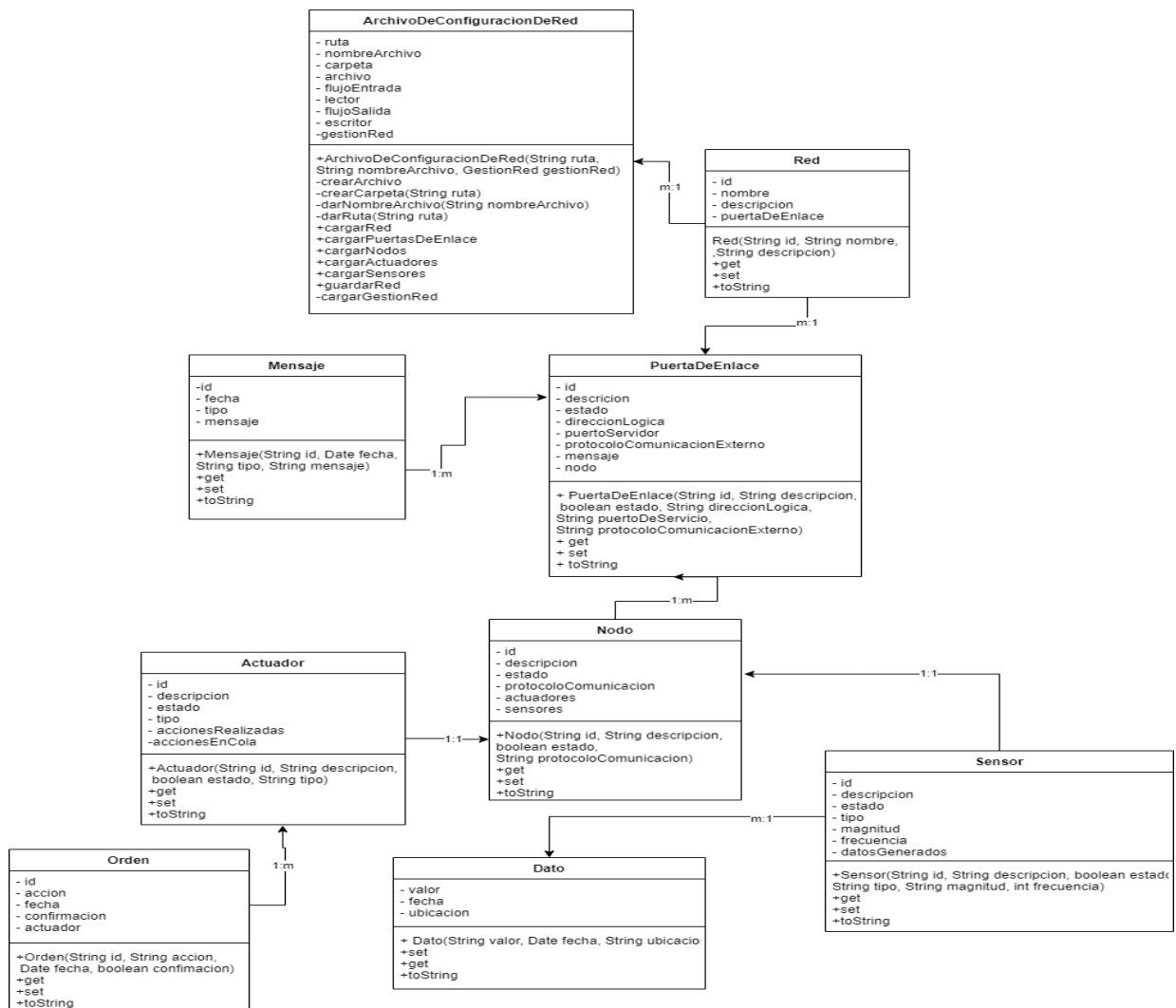


ILUSTRACIÓN 4 DIAGRAMA DE CLASES

Diseño de detalle

La ingeniería de detalle se refiere a los planos, diagramas, modelos, arquitecturas, Mockups o interacciones hombre-máquina de los productos que comprenden la solución presentada en el proyecto.

Arquitectura

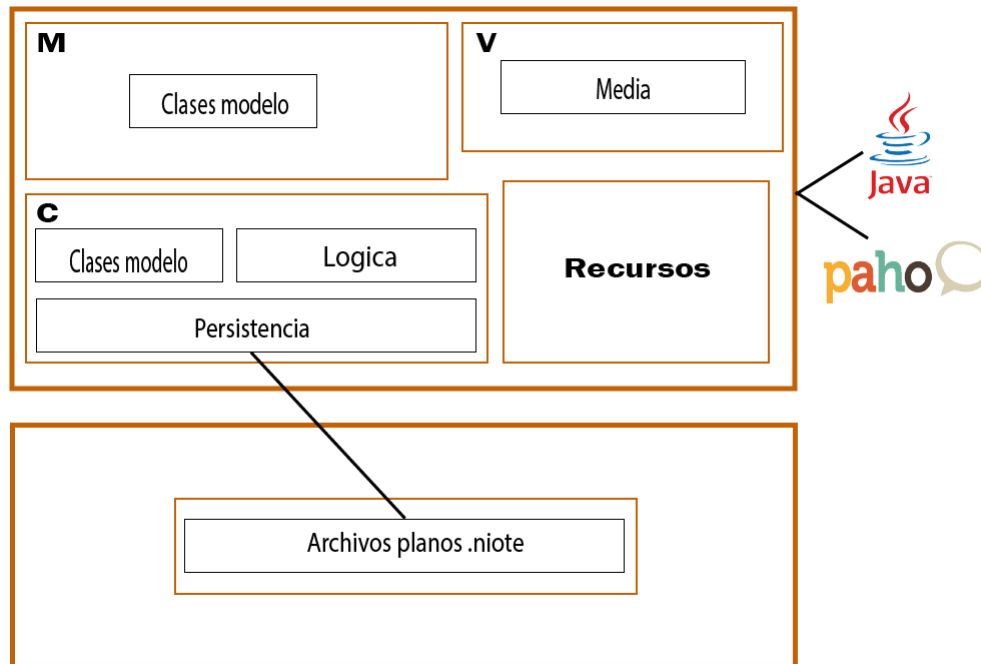


ILUSTRACIÓN 5 ARQUITECTURA DEL PROYECTO

Muckups:

Podemos observar cómo se va a visualizar al usuario cuando utilice el emulador el cual mostrara su respectiva interfaz, con sus respectivos elementos para poder realizar la simulación.

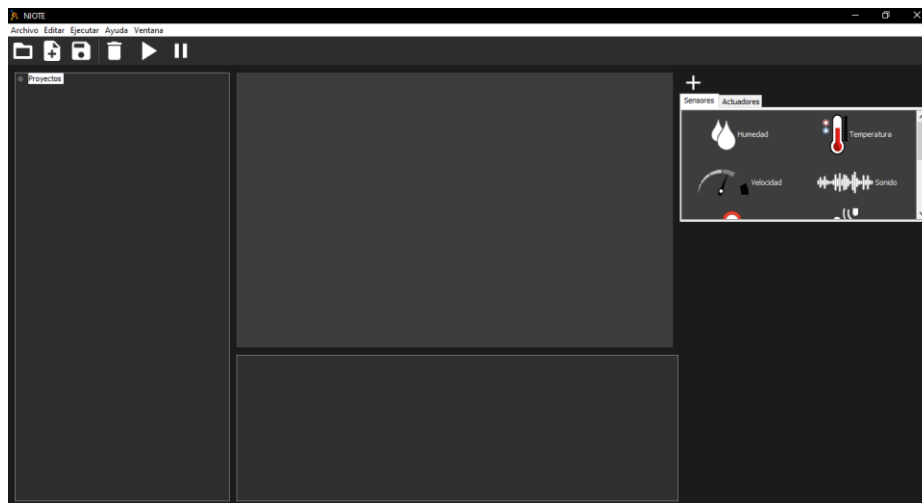


ILUSTRACIÓN 6 PANTALLA PRINCIPAL

El siguiente muestra el panel donde podrá agregar o realizar los diferentes proyectos que contenga una extensión .niote, el cual mostrara en forma de árbol la organización de los proyectos.



ILUSTRACIÓN 7 PANEL DE PROYECTOS

El panel de emulación mostrará los nodos, sensores, actuadores el cual el usuario escoja para realizar su respectiva simulación el cual podrá mover a su antojo los respectivos elementos.

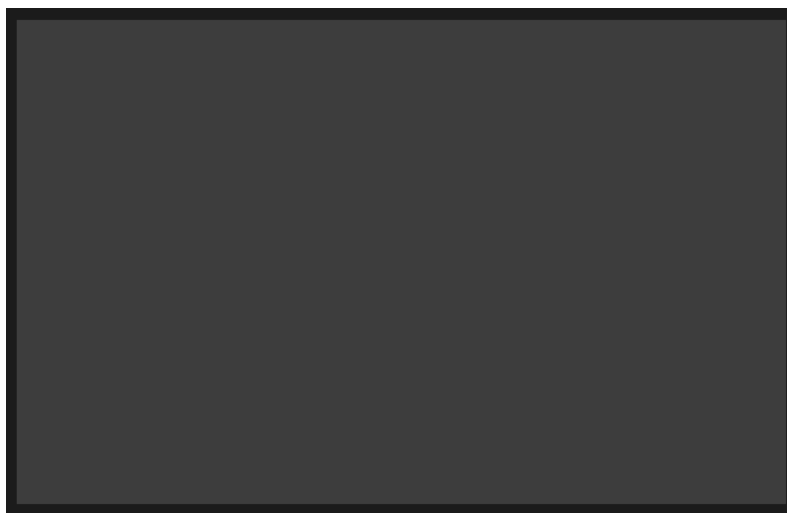


ILUSTRACIÓN 8 PANEL DE EMULACIÓN

Panel output mostrara los elementos seleccionados, su ejecución, mostrara cuando el programa para su ejecución y errores que podrían salir en la simulación de redes IoT,

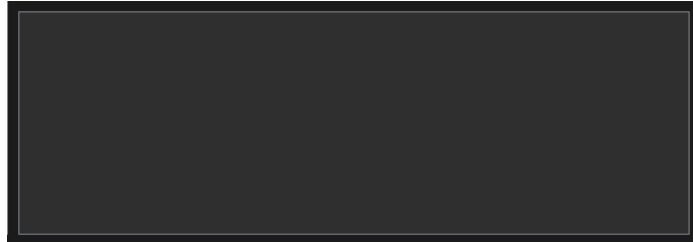


ILUSTRACIÓN 9 PANEL OUTPUT DEL PROYECTO

Panel de Nodos mostrara lo elementos de sensores, actuadores y demás para poder seleccionar y poder mostrarlos en el respectivo panel de emulación



ILUSTRACIÓN 10 PANEL DE NODOS IoT

En este menú bar tendrá las siguientes opciones el cual al seleccionar archivo mostrará unas opciones; la primera opción es la creación de un nuevo proyecto, la segunda podrá abrir un proyecto ya realizado solo permitirá con extensión .niote, la tercera opción podrá cerrar el proyecto que seleccione o el que está utilizando, la cuarta opción cerrara todos los proyectos que estén abiertos en el panel de proyectos la quinta opción podrá exportar la red realizada en el formato png.

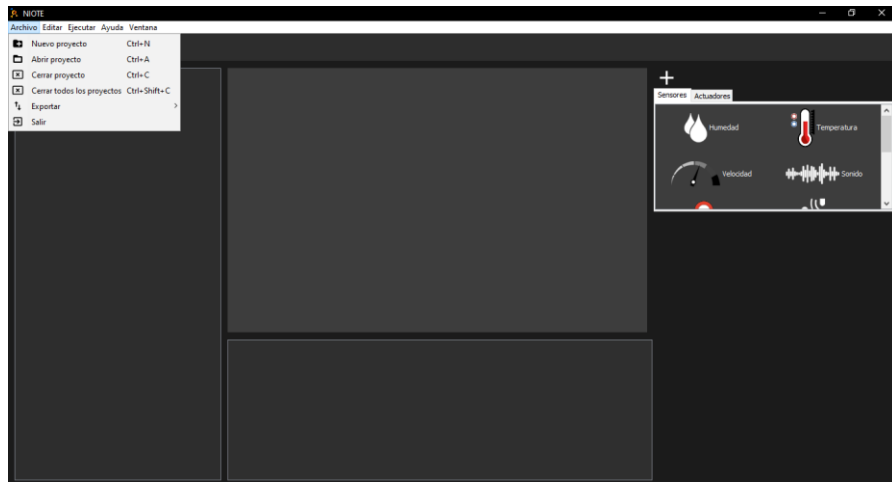


ILUSTRACIÓN 11 NAVEGACIÓN MENUBAR

En el MenuBar tendrá la opción de editar el cual al seleccionarlo solo mostrará una opción el cual será borrar proyecto totalmente.

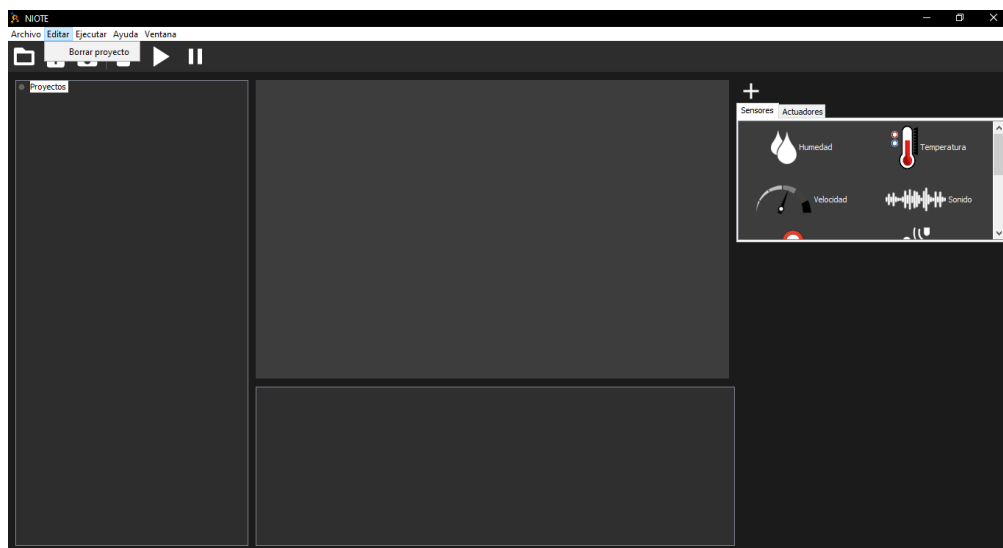


ILUSTRACIÓN 12 NAVEGACIÓN MENUBAR (EDITAR)

En el MenuBar tendrá la opción de ejecutar el cual al seleccionarlo solo mostrará una opción el cual será ejecutar la simulación y también mostrar el comando para realizarlo por el teclado.

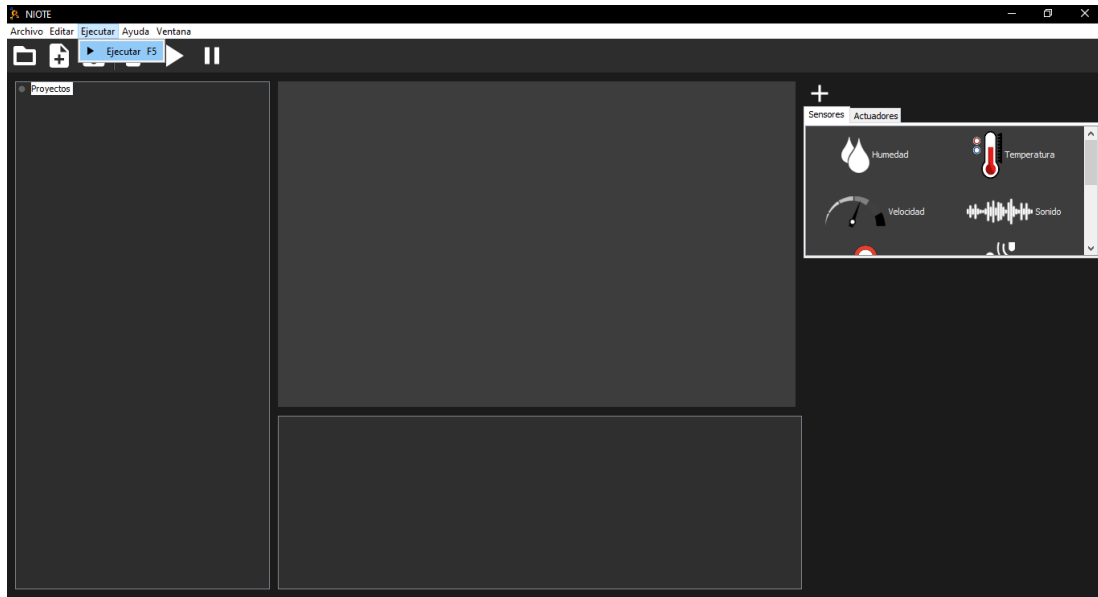


ILUSTRACIÓN 13 NAVEGACIÓN MENUBar (EJECUTAR)

En el MenuBar tendrá la opción de ayuda el cual al seleccionarlo mostrará cuatro opciones la primera es la documentación del usuario el proyecto y funcionamiento, la segunda opción es reportar un erro el cual abrirá una ventana para mandar un mensaje con el error y podrá adjuntar imágenes, la tercera buscará alguna actualización del emulador y la última sería una ventana con la información de los desarrolladores y tutor del proyecto.

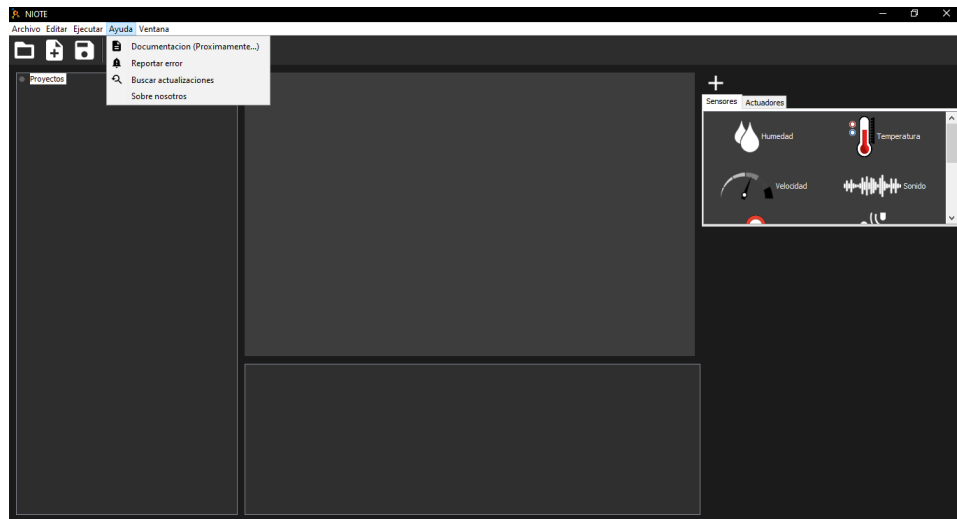


ILUSTRACIÓN 14 NAVEGACIÓN MENUBar (AYUDA)

En la última opción del MenuBar el cual es ventana al seleccionarlo mostrará tres opciones el cual la primera podrá mover la ventana del emulador, el idioma tendrá dos opciones de español e inglés, y, por último, temas el cual será claro u oscuro.

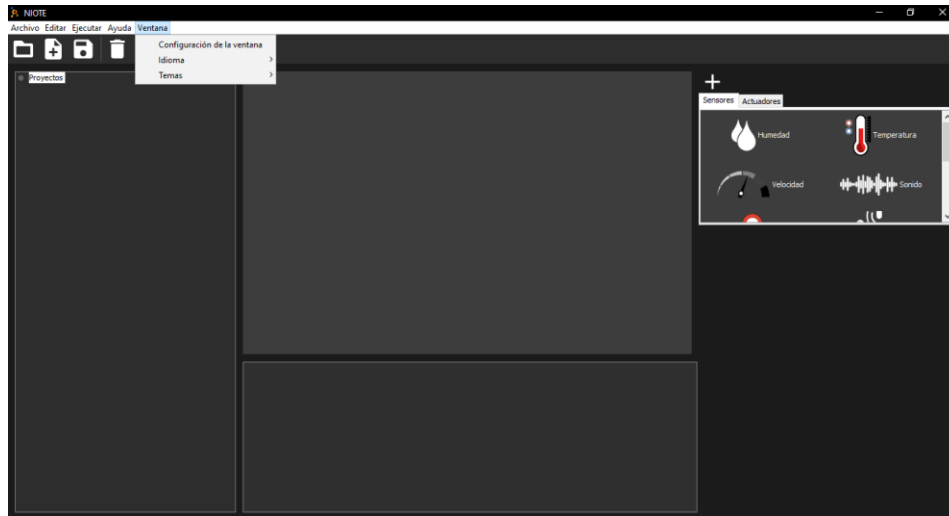


ILUSTRACIÓN 15 NAVEGACIÓN MENUBar (VENTANA)

Proceso de construcción y puesta en marcha.

En este punto, se mostrará todos los elementos usados y requerimientos para el óptimo funcionamiento del emulador.

Entorno de desarrollo

Ahora se evidenciarán todos los elementos necesarios para poder ejecutar el emulador y pueda estar funcionado sin ningún error de ejecución.

- El entorno de desarrollo que utilizamos es Netbeans versión 8,2
- Es necesario tener el JDK versión 8
- Para el funcionamiento del emulador es necesario tener mosquito
- Tener la última versión de java
- Librerías que utilizaremos para el funcionamiento del emulador

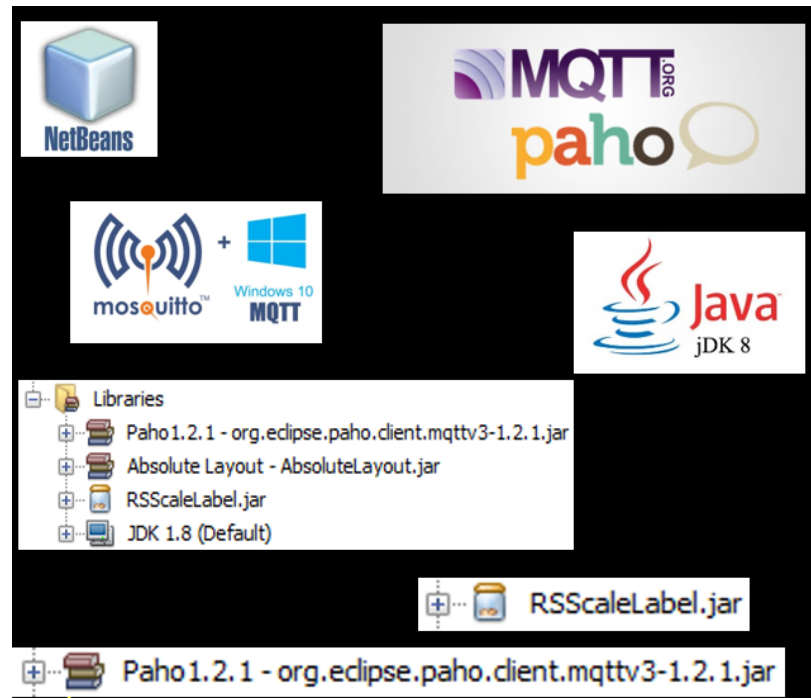


ILUSTRACIÓN 16 ENTORNO DE DESARROLLO

Prueba – Validación

Se diseña e implementa un protocolo de pruebas sobre los productos que comprenden la solución presentada en el proyecto, este protocolo de pruebas va enfocado a validar que los productos cumplen con los objetivos, alcances y limitaciones del proyecto.

En la parte de prueba y validación serializaran sus respectivas pruebas con los avances que se vayan realizando, teniendo un protocolo con cada elemento agregado en el respectivo proyecto, poder validar el cual llevamos un avance con la agregación de los elementos que son necesarios para realizar una red.

En la siguiente imagen se puede evidenciar el funcionamiento de los avances que se han llevado a cabo en el emulador.

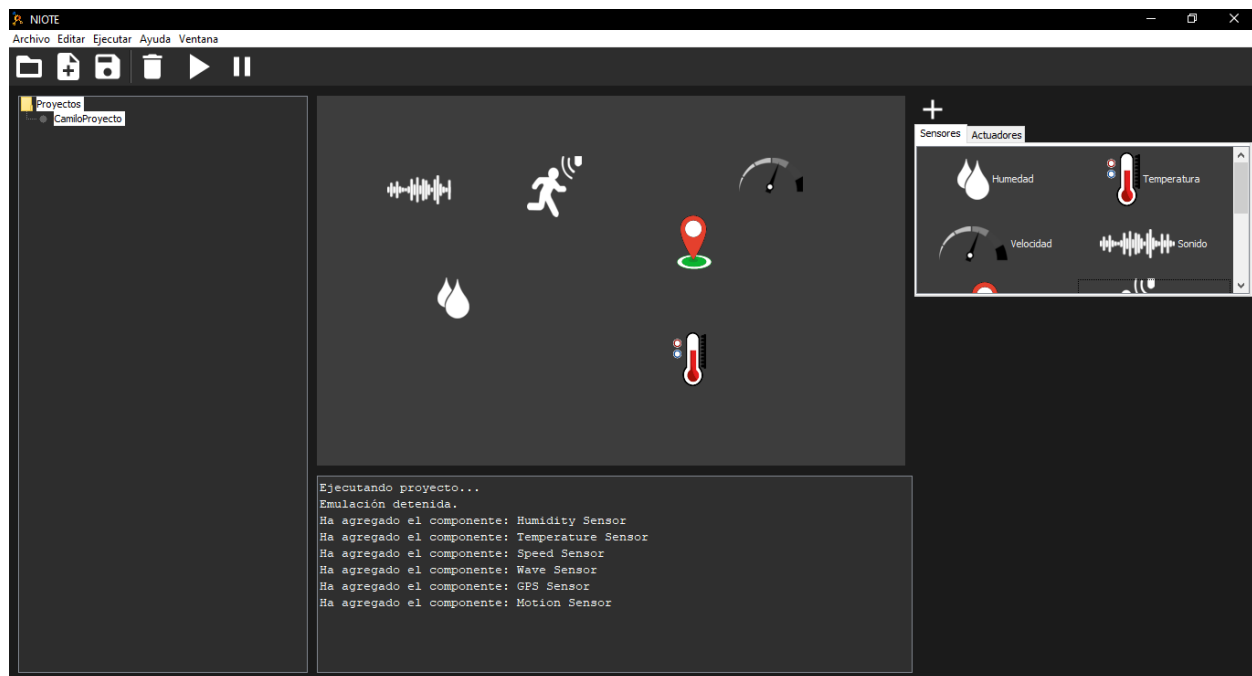


ILUSTRACIÓN 17 PANEL PRINCIPAL FUNCIONANDO

Análisis de Resultados

Ahora, se analizarán los resultados encontrados gracias a la prueba de caja negra que se estableció para la evaluación del FrontEnd del emulador, en este punto, se seleccionó una pequeña población para demostrar los avances y funcionamiento que se obtuvieron en el emulador.

Definición del Caso de Estudio

En el desarrollo de un software uno de los aspectos más importantes a tener en cuenta es la satisfacción del usuario al momento de hacer uso de dicho sistema, por lo tanto, una de las maneras más eficientes para conocer la opinión de los usuarios con respecto al programa es mediante una encuesta que permita conocer y evaluar que tan satisfactorio es el software con respecto al pensamiento de dichos usuarios. De esta manera, se realizó una encuesta donde las preguntas propuestas son basadas a partir de los 10 principios de usabilidad, cada principio fue tomado en cuenta para la evaluación apropiada dirigida a los encuestados. Por lo tanto, en cada principio fueron diseñadas preguntas específicas para evaluar la comodidad y satisfacción por parte de los encuestados con respecto al emulador, con el fin de tomar en cuenta las opiniones dadas por los participantes y mejorar la experiencia proporcionada por el emulador.

Puesta en marcha del Caso de Estudio

Se tomaron en cuenta los avances realizados en el desarrollo a nuestro proyecto que se llevaba en el momento en el que fue desarrollado el FrontEnd del emulador, por lo tanto, nos correspondía realizar las pruebas de caja negra basándonos en los principios de usabilidad.

Teniendo en cuenta lo anteriormente mencionado, en esta prueba se tuvo en cuenta los 10 principios de usabilidad para poder realizar las respectivas preguntas del diseño del emulador; por cada uno de los principios se realizaron preguntas donde los encuestados responderán con base a la escala de Likert; considerando lo anteriormente mencionado estas son las preguntas que se diseñaron para realizar la encuesta a la población seleccionada:

1. Visibilidad del estado del sistema
 - ¿El emulador mantiene informado al usuario de las acciones que realiza?
 - ¿La interfaz visualiza los componentes agregados al panel de simulación con su respectivo nombre?
2. Relación entre el sistema y el mundo real
 - ¿La interfaz del emulador contiene iconos que permiten reconocer al usuario que opción ejerce al ser oprimido?
 - ¿Cada componente como sensores y actuadores son reconocibles gracias a sus iconos?
3. Control y libertad del usuario
 - ¿El emulador permite al usuario corregir o deshacer acciones realizadas en la interfaz?
 - ¿Es posible eliminar los componentes que hayan sido agregados al panel de simulación?
4. Consistencia y estándares
 - ¿Se puede evidenciar algún patrón de diseño gráfico y de navegación en la pantalla principal?

5. Prevención de errores
 - El emulador le permite saber rápidamente si ha cometido algún error
6. Reconocimiento antes que recuerdo
 - El panel principal es fácil de recordar para próximos usos a futuro del emulador, es decir, sus atajos, la navegación, entre otros.
7. Flexibilidad y eficiencia de uso
 - Los atajos de teclado son fáciles de recordar.
 - Los atajos me ayudan en la navegación del emulador.
8. Estética y diseño minimalista
 - ¿La cantidad de paneles que se encuentran en el panel principal son suficientes?
 - ¿Hay saturación de información en el panel principal?
 - La navegación es fácil e intuitiva a primera vista.
 - ¿Los colores para el tema oscuro son adecuados?
 - ¿La distribución de los paneles está bien establecida?
 - La interfaz es agradable.
 - El icono y logo de la aplicación es agradable.
9. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores
 - El panel de salida, el que se encuentra en la parte inferior, es claro y simple a la hora de reportar las acciones y salidas de la simulación en ejecución
10. Ayuda y documentación
 - Es fácil encontrar la documentación para resolver dudas.
 - La documentación dada es clara y concisa con la solución de sus dudas.

Teniendo en cuenta las preguntas y las respuestas por parte de los usuarios encuestados se realizó un análisis para determinar el nivel de satisfacción en la parte visual del emulador.

Análisis estadístico/probabilístico

Como se menciona en la puesta en marcha del caso de estudio se realizaron pruebas de caja negra donde se obtuvieron respuestas relevantes para el análisis de estas gracias a las preguntas realizadas. Por lo tanto, se analizarán cuatro de las preguntas generadas, las cuales, son las mas sobresalientes para el análisis de los resultados.

Teniendo en cuenta lo anterior, se seleccionaron las siguientes preguntas donde también se evidenciará sus respectivos resultados:

La interfaz del emulador contiene iconos que permiten reconocer al usuario que opción ejerce al ser oprimido.
17 respuestas

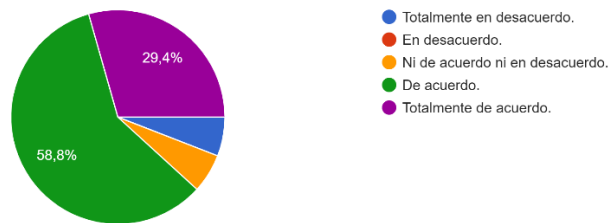


ILUSTRACIÓN 18 RESPUESTAS CON RESPECTO A LA VISIBILIDAD DEL ESTADO DEL SISTEMA Y LA ÓPTIMA APLICACIÓN DE LOS ICONOS SELECCIONADOS

La interfaz es agradable.
17 respuestas

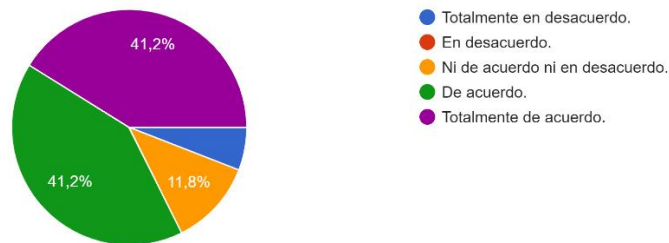


ILUSTRACIÓN 19 RESPUESTAS CON RESPECTO A LA ESTÉTICA Y DISEÑO MINIMALISTA Y EL GUSTO POR LOS ENCUESTADOS A LA INTERFAZ

La distribución de los paneles esta bien establecida.
17 respuestas

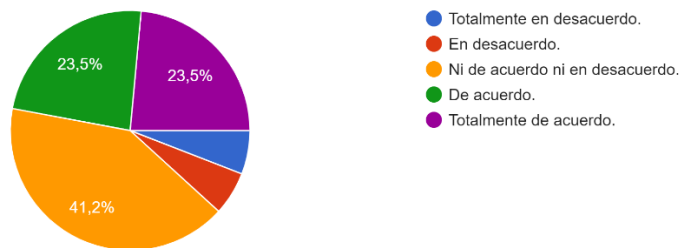


ILUSTRACIÓN 20 RESPUESTAS CON RESPECTO A LA ESTÉTICA Y DISEÑO MINIMALISTA Y LA ÓPTIMA DISTRIBUCIÓN DE LOS PANELES EN LA INTERFAZ PRINCIPAL

Los colores para el tema oscuro son adecuados.
17 respuestas

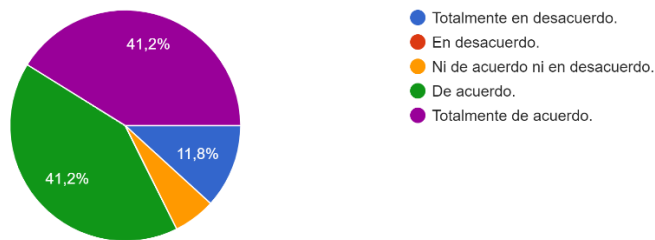


ILUSTRACIÓN 21 RESPUESTAS CON RESPECTO A LA ESTÉTICA Y DISEÑO MINIMALISTA Y LA BUENA SELECCIÓN DE COLORES PARA EL TEMA OSCURO EN LA INTERFAZ PRINCIPAL

Se tiene que resaltar que en las contestaciones se encuentran las de un ingeniero de sistemas, por lo tanto, sus respuestas van a tener mas peso en los resultados ya que es la única persona que a usado IDE's y programas de emulación y simulación. Considerando lo mencionado anteriormente, las respuestas mencionadas por el ingeniero encuestado son "Totalmente de acuerdo" en todas las preguntas seleccionadas para el análisis.

Ahora, teniendo en cuenta los resultados se puede evidenciar que un 82% de los encuestados piensan que la interfaz es agradable resaltando el tema oscuro seleccionado ya que a muchos de ellos les agrado los colores escogidos, los iconos aplicados para cada ítem, pero, aun así, a un 41% de los encuestados quienes son la mayoría no están ni de acuerdo ni en desacuerdo en la distribución de los paneles que componen la interfaz principal; se tiene que resaltar que la mayoría de ellos no han usado IDE's los cuales tienen una interfaz y distribución similar, ahora bien, se tiene que destacar la respuesta del ingeniero encuestado, el cual, está totalmente de acuerdo en que la distribución de los paneles está bien establecida ya que el si ha usado IDE's para sus estudios y vida laboral.

Conclusiones

Teniendo en cuenta que el emulador hasta este punto va su desarrollo en el FrontEnd, las conclusiones serán basadas en este ámbito ya que no se ha realizado el BackEnd ni su distribución para el análisis de la efectividad del emulador.

Conclusiones

- El emulador permite a los usuarios ahorrar tiempo y recursos a la hora de desarrollar una red IoT.
- El emulador admite crear simulaciones de redes IoT de manera fácil y rápida.
- El emulador otorgara una simulación muy acertada a la realidad, con la cual, los usuarios podrán fiarse de esta.

Lecciones aprendidas y experiencia

Se presentaron varios inconvenientes en la construcción de la parte visual del emulador ya que se encontraron errores en la estructuración y programación de este; aun así, fueron satisfactoriamente superados con una ardua investigación, la cual, generó y amplio conocimiento en programación con Java y uso del sistema operativo para complementar funciones que se generaron en el FrontEnd.

Trabajos futuros

El emulador se construyó la parte visual, es decir, el FrontEnd, el cual, no está desarrollado en su totalidad y esta es la razón por la que se continuara con el desarrollo del emulador, donde posteriormente, se desarrollara el BackEnd para la finalizar la construcción del emulador y observar el uso e impacto que generara a los usuarios que vayan a usarlo en un futuro.

Bibliografía y Referencias

- Semle, A. (Septiembre de 2016). Protocolos IIoT para considerar. *AADECA Revista*, 32-35. Recuperado el 25 de Marzo de 2020, de https://editores-srl.com.ar/sites/default/files/aa2_semle_protocolos_iiot.pdf
- Ahrenholz, J., Danilov, C., Herderson, T., & Kim, J. (2008). A real-time network emulator. In *Military Communications Conference. MILCON*, 1-7.
- Amine Khelif, M., Lorandel, J., Romain, O., Regnery, M., & Baheux, D. (2019). A Versatile Emulator of MitM for the identification of vulnerabilities of IoT devices, a case of study: smartphones. En A. f. Machinery (Ed.), *3rd International Conference on Future Networks and Distributed Systems*, (págs. 1-6). Paris. Recuperado el 2 de Abril de 2020, de <https://dl.acm.org/doi/pdf/10.1145/3341325.3342019>
- AprendiendoArduino. (17 de Noviembre de 2018). *Protocolos IoT Capa Aplicación*. Recuperado el 14 de Abril de 2020, de Aprendiendo Arduino: <https://aprendiendoarduino.wordpress.com/2018/11/17/protocolos-iot-capa-aplicacion/>
- arrow. (1 de Julio de 2015). *Protocolos para la Internet de las cosas*. Recuperado el 17 de Abril de 2020, de arrow: <https://www.arrow.com/es-mx/research-and-events/articles/protocols-for-the-internet-of-things>
- CambioDigital. (12 de diciembre de 2018). Recuperado el 13 de marzo de 2020, de IoT: Qué necesitan saber los profesionales de la red: <https://cambiodigital-ol.com/2018/12/iot-que-necesitan-saber-los-profesionales-de-la-red/>
- Carlos Gamero Burón, J. L. (2015). *Modelos probabilísticos para Variables aleatorias continuas*. Malaga, España.
- Castellanos Hernández, W. E., & Chacon Osorio, M. E. (17 de Abril de 2006). Utilización de herramientas software para el modelado y la simulación de redes de comunicaciones. *GTI*, V(11), 74-75. Recuperado el 26 de Marzo de 2020, de <https://revistas.uis.edu.co/index.php/revistagti/article/view/1624/2014>
- Corso, C., & Lorena, C. (2009). *Aplicacion de algoritmos de clasificacion sepervisan y no supervisada usando Weka*. cordoba: Universidad Tecnologi Nacional.
- Coss Bu, R. (2003). *Simulación un enfoque practico*. Monterrey, Mexico: Limusa. Recuperado el 25 de Marzo de 2020, de <https://books.google.es/books?hl=es&lr=&id=iY6dl3E0FNUC&oi=fnd&pg=PA111&dq=simulacion&ots=uKV85h0Scu&sig=fMdlmFTXdSYn3HghHIZ7HbFXQhg#v=onepage&q=simulacion&f=false>
- Crespo Moreno, J. E. (11 de Noviembre de 2018). *Aprendiendo Arduino*. Recuperado el 2 de Abril de 2020, de Arquitecturas IoT: <https://aprendiendoarduino.wordpress.com/2018/11/11/arquitecturas-iot/>

- cuatro tipos de analítica de retail que todo comercio necesita en 2018. (7 de Agosto de 2018). *Analítica de retail*. Obtenido de Analítica de retail: <http://analiticaderetail.com/tipos-de-analitica-de-retail/>
- Diaz, T., Escriba, F., & Murgui, M. (2002). La base de datos BD. MORES. *Revista de Economía Aplicada*, 165-184.
- Dunkles, A., Schmidt, O., Finne, N., Erikson, J., Osterlind, F., Tsiftes, N., & Durvy, M. (2011). *The contiki os: The operating system for the internet of things*. Obtenido de Online: <http://www.contikios.org>
- editorial, E. (1 de Noviembre de 2018). *REPORTEGIGITAL*. Obtenido de REPORTEGIGITAL: <https://reportedigital.com/cloud/analitica-de-datos/>
- Eduardo García Dunna, H. G. (2013). *Simulación y análisis de sistemas con ProModel*. Naucalpan de Juárez, Estado de México, México: PEARSON. Obtenido de <https://jrvargas.files.wordpress.com/2015/04/libro-simulacion-y-analisis-de-sistemas-2da-edicion.pdf>
- Gan, S. (2017). *An IoT simulator in NS3 and a key-based authentication architecture for IoT devices using blockchain*. Tesis, Instituto Indio de Tecnología Kanpur, Ciencias informáticas e ingeniería, Kanpur. Recuperado el 2 de Abril de 2020, de https://security.cse.iitk.ac.in/sites/default/files/12807624_0.pdf
- García Sánchez, Á., & Ortega Mier, M. (2006). *Introducción a la simulación de sistemas discretos*. Recuperado el 25 de Marzo de 2020, de http://www.iol.etsii.upm.es/arch/intro_simulacion.pdf
- Garcia, P. (2006). *TECNICAS DE ANALITICA*.
- Gibbs, G. (2012). *Análisis de datos en investigaciones cualitativas*. Ediciones Morata.
- Gomez-Aguilar, D., Garcia-Peñalvo, F., & Theron, R. (2014). Analítica visual en learning. *El profesional de la informática*, 23(3).
- Hassan, F., Domingo-Ferrer, J., & Soria-comas, J. (2018). Anominaion de datos no estructuados a traves del reconocimiento de entidades nominadas. *Actas de la XV Reunni Espaola sobre Criptologa y Seguridad de la informcin-RECSI*, 102-106.
- Hergert, M., & Morris, D. (1989). Datos contables para el analisis de la cadena de valor. *Diario de gestion estrategica*, 10(29),175-188.
- Hernandez Perez, A. (2013). *Datos abiertos y repositorios de datos*. nuevo reto para los bibliotecarios.
- Hernandez, C., & Rodriguez, J. (2008). Preprocesamiento de datos estructurados. *Vinculos*, 27-48.
- Huang, Y., Wang, L., Hou, Y., Zhang, W., & Zhang, Y. (2018). *A prototype IOT based wireless sensor network for traffic information monitoring*. *International Journal of Pavement Research & Technology*.
- Isaac Lera, C. G. (2019). *YAFS*. Palma. Obtenido de <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8758823>

- itop. (20 de agosto de 2018). Recuperado el 13 de marzo de 2020, de IoT: origen, importancia en el presente y perspectiva de futuro: <https://www.itop.es/blog/item/iot-origen-importancia-en-el-presente-y-perspectiva-de-futuro.html>
- Joyanes Aguilar, L. (29 de mayo del 2019). *Inteligencia de negocios y analítica de datos*. Bogotá: Alfaomega.
- Leónardo Darío Bello Parías, L. C. (2000). *Libro de estadística descriptiva*. Medellín, Antioquia, Colombia: Editorial Amistad ISBN. Obtenido de http://aprendeenlinea.udea.edu.co/lms/moodle/pluginfile.php/128508/mod_resource/content/0/Tema_4/Modelos_probabilisticos_Caucasia.pdf
- Levis, P., Lee, N., Welsh, M., & Culler, D. (2003). *TOSSIM: Accurate and scalable simulation of entire Tinyos applications*. In Proceedings of the 1st international conference on Embedded networked sensor systems.ACM.
- Llamas, L. (2019). *Protocolos de comunicación para IoT*. Recuperado el 17 de Abril de 2020, de Luis Llamas: <https://www.luisllamas.es/protocolos-de-comunicacion-para-iot/>
- Loukides, M. (2011). *¿Que es la ciencia de datos?* O'Reilly Media, Inc.
- Mäkinen, A. (2016). *Emulation of IoT Devices*. Espoo. Obtenido de https://aaltodoc.aalto.fi/bitstream/handle/123456789/23951/master_M%c3%a4ki_nen_Alli_2016.pdf?sequence=1&isAllowed=y
- Mehmood, T. (s.f.). *COOJA Network Simulator*. Islamabad. Obtenido de <https://arxiv.org/ftp/arxiv/papers/1712/1712.08303.pdf>
- Myriam Muñoz de ózak, S. F. (1991). PROCESOS ESTOCÁSTICOS CON DOS PARÁMETROS I. 72-74. Obtenido de <https://revistas.unal.edu.co/index.php/estad/article/viewFile/9955/10486>
- NEO.LCC. (s.f.). *Protocolos de transporte*. Recuperado el 14 de Abril de 2020, de Herramientas WEB para la enseñanza de protocolos de comunicación: <http://neo.lcc.uma.es/evirtual/cdd/tutorial/transporte/protrans.html>
- Nigro, O., Corti, D., & Terren, D. (2004). *Knowledge Discovery in Databases*. Un proceso centrado en el usuario. In VI Workshop de investigadores en Ciencias de la Computación.
- Node-Red. (2013). *Low-code programming for event-driven applications*, 1.0.5. Obtenido de Node-Red: <https://nodered.org/>
- NSNAM. (2011). *NS-3 Network Simulator*. Recuperado el 2 de Abril de 2020, de NS-3: <https://www.nsnam.org/>
- Quintero, J. (2006). La cadena de valor : Una herramienta de pensamiento estratégico. Telos, 14.
- Quiñones Cuenca, M., González Jaramillo, V., Torres, R., & Miguel, J. (2017). *Sistema de Monitoreo de variables medioambientales usando una red de sensores*

inalámbricos y plataformas de Internet de las Cosas. Universidad Técnica Particular de Loja, Departamento de Ingeniería, Loja. Recuperado el 14 de Abril de 2020, de http://scielo.senescyt.gob.ec/scielo.php?script=sci_arttext&pid=S1390-65422017000100329

Raposo, J. (2007). *Tecnicas de mantenimiento automatico de programas envoltorio para fuentes de datos web semiestructuradas*. Coruña: Doctoral dissertation.

Riverder Technologies. (2017). *opnet simulator*. Obtenido de <https://www.riverbed.com/in/products/steelcentral/opnet.html>

Rodríguez Moreno, E. S., & López Ordoñez, V. F. (2017). *Diseño e implementación de un sistema inteligente para un edificio*. Tesis de grado, Universidad Francisco Jose de Caldas, Facultad de Ingeniería, Bogotá. Recuperado el 25 de Marzo de 2020

Roldán Carrasco, Á. (2007). *Emulador de Gameboy para dispositivos móviles*. Tesis, Escuela Superior de Ingeniería Informática, Departamento de Informática, Ciudad Real. Recuperado el 25 de Marzo de 2020, de <https://www.esi.uclm.es/www/cglez/downloads/pfc/pfcaroldan.pdf>

Sánchez Martín, A. A., Barreto Santamaría, L. E., Ochoa Ortiz, J. J., & Villanueva Navarro, S. E. (2019). *Emulador para desarrollo de proyectos IoT y analíticas*. PREGUNTAR, Universidad Distrital Francisco José de Caldas, Bogotá. Recuperado el 13 de marzo de 2020

SimpleSoft. (s.f.). *SimpleSoft*. Obtenido de <http://www.smplsft.com/SimpleIoT Simulator.html>

Snoke, J., & HalanLarochelle. (2012). *Practica y optimizacion de algoritmos de aprendizaje automatico. Avances en sistemas de procesamiento de informacion neuronal*, 2951-2959.

Tetcos. (2017). *Netsim emulator*. Obtenido de <http://tetcos.com/>

Torres Bataller, J. (2016). *Desarrollo de una solucion para la simulacion de entornos IoT*.

Universidad de Alcalá. (2019). *¿Por qué actualmente es tan importante el IoT?* Recuperado el 13 de marzo de 2020, de Máster en industria 4.0: <https://www.masterindustria40.com/importancia-iot-master/>

Varga, A. (2016). *In Modeling and tools for network simulation*. Berlin,Heidelberg: Springer.

Xia, F., Yang, L., Wang, L., & Vinel, A. (2012). *Internet of Things*. International journal of communication systems. doi:10.1002/dac.2417

Yacchirema Vargas, D. C., & Palau Salvador, C. E. (s.f.). *Smart IoT Gateway For Heterogeneous Devices Interoperability* (Octava ed., Vol. 14). IEEE Latin America Transactions. doi:10.1109/TLA.2016.7786378