Propuesta de arquitectura para el desarrollo e implementación de un módulo de nuevos comandos e instrucciones para Google Assistant

[a] *I.S.C Noé Hernández García.,* [b] *M.R.T. Ignacio López Martínez.,* [c] *M.C.E. Beatriz A. Olivares Zepahua.,* [d] *M.C. S. Gustavo S. Peláez Camarena.,* [e] *M.C. Ma. Antonieta Abud Figueroa*

Instituto Tecnológico de Orizaba, División de Estudios de Posgrado e Investigación, Maestría en Sistemas Computacionales.
[a] noe-hdezg@hotmail.com, [b] ilopez@ilopezm@orizaba.tecnm.mx, [c] bolivares@ito-depi.edu.mx, [d] sgpelaez@yahoo.com.mx, [e] mabud@ito-depi.edu.mx, Orizaba, Veracruz, Mexico.

Resumen

Actualmente, la domótica es un área de mucho interés e innovación, ya que su objetivo principal es la automatización de diversas tareas en el hogar. Para lograr un ambiente domótico, es necesaria la intervención de ciertos dispositivos especializados, como los asistentes virtuales, que no son más que agentes computarizados encargados de realizar varias acciones con muy poca interacción de los usuarios, dicha interacción es mayormente llevada a cabo por comandos de voz que manda el usuario al dispositivo, como en el caso del *Google Home Mini*. Gracias a la realización de un estudio relacionado con el tema de los asistentes virtuales, se llegó a la conclusión de que existe muy poca información con respecto a la creación de nuevos comandos de voz para asistentes virtuales, y no existe una guía o una pauta que de paso a la creación de estos. Por lo que además de conocer las características de las tecnologías antes mencionadas, se llegó a la presente propuesta: generar una arquitectura que sea utilizada en la creación de nuevos comandos de voz para el *Google Assistant.* Dicho asistente virtual está integrado en el *Google Home Mini,* mismo que será utilizado para probar los nuevos comandos de voz resultantes de la utilización de la nueva arquitectura, además, se propone la utilización de *Dialogflow* y el SDK (*Software Development Kit,* Paquete de Desarrollo de Software) de Google Assistant.

Palabras clave— Google Assistant, Asistentes Virtuales, Lenguaje natural, Procesamiento de lenguaje natural, SDK Google.

Abstract

Currently, domotics is an area of great interest and innovation, since its main objective is the automation of various tasks in the home. To achieve a domotic environment, it is necessary the intervention of certain specialized devices, such as virtual assistants, which are no more than computerized agents responsible for carrying out various actions with very little user interaction, such interaction is mostly carried out by voice commands sent by the user to the device, as in the case of Google Home Mini. Thanks to a study related to the topic of virtual assistants, it was concluded that there is very little information regarding the creation of new voice commands for virtual assistants, and there is no guide or guideline that leads to the creation of these. So in addition to knowing the characteristics of the technologies mentioned above, we came to the present proposal: to generate an architecture to be used in the creation of new voice commands for Google Assistant. This virtual assistant is integrated in the Google Home Mini, which will be used to test the new voice commands resulting from the use of the new architecture, in addition, the use of Dialogflow and the SDK (Software Development Kit) of Google Assistant is proposed.

Keywords*— Google Assistant, Virtual Assistants, Natural Language, Natural Language Processing, Google SDK.*

1. INTRODUCCIÓN

Los asistentes virtuales inteligentes según [1], son un conjunto de programas informáticos capaces de interactuar con los seres humanos en su propio lenguaje, esto quiere decir que son entidades autónomas que ofrecen colaboración a las personas para alcanzar objetivos a través de diferentes canales de interacción. Los asistentes virtuales se pueden clasificar de la siguiente manera: aquellos que tienen un dispositivo físico y los que se encuentran embebidos en los sistemas móviles.

Actualmente es posible configurar dichos asistentes para ejecutar más acciones de las que poseen, sin embargo, la revisión de los trabajos relacionados demostró que no existe una arquitectura para construir dichas acciones, por lo que el objetivo de este trabajo es proponer una arquitectura que sirva de guía para la programación de nuevas funcionalidades para el asistente virtual de Google, siguiendo una metodología de desarrollo.

Para lograr dicha premisa, se hizo un estudio previo con el fin de seleccionar las tecnologías necesarias para el desarrollo e implementación de comandos e instrucciones para el asistente virtual Google.

En el Marco Teórico, se encuentran los conceptos y definiciones necesarias para entender el alcance de este proyecto, mientras que en la sección de Trabajos Relacionados, se exponen los artículos relacionados al tema, en el Proceso de Desarrollo, se explica directamente el proceso de desarrollo de la metodología propuesta en este documento, misma que será detallada en la sección de Desarrollo de la Arquitectura; finalmente, en la parte de Conclusiones y Recomendaciones, se limitan los alcances de la metodología y, en Trabajos a Futuro, se plantea la aplicación de dicha metodología en un caso de estudio para probar su utilidad y aplicación.

2. MARCO TEÓRICO

2.1 Domótica

La integración tecnológica de los sistemas electrónicos en el hogar se conoce como domótica. En [2] se define este término como agrupaciones automatizadas de equipos, normalmente

asociados por funciones, que disponen de la capacidad de comunicarse interactivamente entre ellos a través de una conexión multimedia que los integra.

2.2 Lenguaje natural

El lenguaje natural es el conjunto de signos y símbolos orales por medio de los cuales los seres humanos se comunican entre sí, a través de la formación de palabras que expresan ideas [3].

El uso del lenguaje natural en los asistentes virtuales es una pieza clave en el desarrollo de nuevos comandos e instrucciones para el *Google Assistant*, ya que es la forma en que el usuario interactúa con este último; si bien la complejidad de identificar, más que entender dicho lenguaje está cubierta por el propio asistente virtual, aún existen adecuaciones a la forma de expresarse que no se interpretan de forma correcta.

2.3 Procesamiento de lenguaje natural

El Procesamiento del Lenguaje Natural (PLN) es una transformación del lenguaje natural a una representación formal, dicha representación se manipula y de ser necesario, sus resultados se llevan nuevamente al lenguaje natural.

El PLN incluye la recuperación y extracción de información, traducción automática, sistemas de búsquedas de respuestas, generación de resúmenes automáticos, minería de datos y análisis de sentimientos [4].

2.4 Agentes inteligentes

Un agente es una entidad autónoma capaz de almacenar conocimiento sobre sí misma y sobre su entorno, de otra forma, es un programa que, basándose en su propio conocimiento, realiza un conjunto de operaciones para satisfacer las necesidades de un usuario o de otro programa, bien por iniciativa propia o porque alguno de los anteriores se lo ordena [5]. El agente inteligente de *Google*, es capaz de analizar el lenguaje natural y obtener las palabras clave que sirven para ejecutar una acción o instrucción.

2.5 Google Assistant

El asistente virtual personal de *Google* es una tecnología desarrollada con inteligencia artificial, su principal característica es la conversación bidireccional entre los usuarios y *Google*, donde las preguntas y respuestas se pueden complementar sin restringirse a un solo tema [6]. Brinda la posibilidad de comunicación bidireccional en forma de una conversación real. El usuario puede interactuar con *Google Assistant* (GA) con la ayuda de comandos de voz gracias a sus algoritmos de procesamiento de lenguaje natural, además, realizar búsquedas en Internet, programar citas, configurar alarmas y hasta administrar la capa de sesión de medios, también se encuentra como un servicio disponible en la nube. Además, *Google Assistant*, brinda posibilidades de incrementar las funciones o acciones que permitan alcanzar un mayor grado de interacción con los usuarios, para ello se dispone de un entorno de desarrollo proporcionado por el propio *Google*.

2.6 SDK de Google Assistant

El SDK de *Google Assistant* es una tecnología que permite agregar nuevos comandos de voz a través de la detección de palabras clave y mediante la comprensión del lenguaje natural y el uso de la inteligencia de *Google* para recibir ideas.

El servicio de *Google* expone una API de bajo nivel que le permite manipular directamente los bytes de audio de una solicitud y respuesta del Asistente. Se pueden generar enlaces para esta API en lenguajes como Node.js, Go, C ++ y Java para todas las plataformas que admiten gRPC, este un moderno framework de código abierto y alto rendimiento que puede funcionar en cualquier entorno.

Primero se captura una solicitud de audio hablada, ésta se envía al asistente y posteriormente se recibe una respuesta de audio hablada, además del texto en bruto de la emisión [7].

Anteriormente para llevar a cabo una acción deseada, se debía hacer uso de árboles conversacionales, los cuales permitían mostrar la interacción del lenguaje natural con el SDK de Google; estos árboles conversacionales eran transmitidos mediante un formato de texto denominado JSON, el cual se ayudaba de toda la gama de lenguajes de programación que admite gRPC.

3. TRABAJOS RELACIONADOS

En [8] se estableció que el uso de asistentes inteligentes y la automatización de hogares inteligentes se vuelven cada vez más fuerte. Los asistentes virtuales habilitados para voz ofrecen una amplia variedad de servicios orientados a la red y, en algunos casos, logran conectarse a entornos inteligentes, mejorándolos con interfaces de usuario nuevas y efectivas, sin embargo, dichos dispositivos revelan nuevas necesidades y debilidades, por ello, se desarrolló un nuevo asistente inteligente con la capacidad de observar un rostro y determinar las emociones de este; dicho asistente inteligente se ejecuta en un dispositivo Raspberry Pi 3 por medio de una interfaz gráfica desarrollada en HTML5 que contiene JavaScript necesario para comunicarse con una conexión Websocket y que implementa la API (*Application Programming Interface*, Interfaz de Programación de Aplicaciones) RESTful con los módulos de síntesis y reconocimiento de voz, esto permite una mejor interacción entre el usuario y el asistente inteligente.

Debido a que los comandos de voz se encuentran expuestos a diversas situaciones que alteran el entendimiento del lenguaje natural en los dispositivos, Michaely et al. [9] presentó un nuevo sistema de localización de palabra clave, este

sistema que utiliza el ASR (*Automatic Speech Recognition*, Reconocimiento Automático de Voz) del lado del servidor y una frase desencadenante para mejorar la precisión general de KWS (*Knowledge work system*, Sistemas de Manejo de Conocimiento), el cual, permite al usuario hablar sin interrupciones entre la frase de activación y el comando de voz, reduciendo así significativamente la tasa de falsas aceptaciones (FA) en un 89%, al mismo tiempo que aumenta de manera mínima la tasa de falsos rechazos (FR) en un 0.2% para mejorar la calidad de ASR.

En [10] C. Peng y R. Chen desarrollaron una función hecha a medida para los usuarios, el cual consiste en el encendido y apagado de un socket, utilizando el reconocimiento de voz de *Google Home* con la concepción del aprendizaje automático para probar el análisis de viabilidad sobre cómo satisfacer las necesidades de los usuarios mediante un patrón de hogar inteligente con el diseño del aprendizaje automático utilizando LSA (*Latent Semantic Analysis*, Análisis semántico latente) and TF-IDF (*Term Frequency–Inverse Document Frequency*, Frecuencia de Término – Frecuencia Inversa de Documento). Para conectar el dispositivo móvil con el *Raspberry Pi* se utilizó BluePy que es una suite que proporciona API (*Application Programming Interface*, Interfaz de Programación de Aplicaciones) para conectar *Bluetooth* bajo consumo basados en el lenguaje *Python* el cual se conecta con el dispositivo *Google Home*, el *Bluetooth* abre una conexión utilizando API.ia hoy conocido como *Dialogflow* permitiendo conversaciones en lenguaje natural humano-computadora mediante su SDK, este permitirá la creación de la nueva función. El experimento permite a los usuarios enviar comentarios al reconocimiento de voz de *Google Home* y luego transferir la señal de *Bluetooth* a *Raspberry Pi* para controlar los dispositivos.

Hwang et al. [11] propusieron una arquitectura para la generación automática de guías de interacción de usuario con asistente inteligente. Dado que el mayor obstáculo es la dificultad que tienen los usuarios conocer el alcance del servicio (característica o conocimiento) y, por lo tanto, es difícil entregar la solicitud correcta. Si el asistente inteligente sugiere las funciones adecuadas basadas en el contexto o las preferencias del usuario con el idioma o las voces naturales, podría hacer que los usuarios estén familiarizados con él, por lo que podría hacer que los asistentes inteligentes sean muy utilizados. Para el desarrollo de esta arquitectura se utilizó una ontología denominada red de tareas, esta red de tareas se usa para encontrar la secuencia correcta de acciones para lograr la solicitud del usuario. Se utilizan los métodos NLG/TTS (*Natural Language Generation/Text-To-Speech*, Generación de Lenguaje Natural/Texto A Voz) para hacer la guía natural basada en los planes que el sistema desea recomendar.

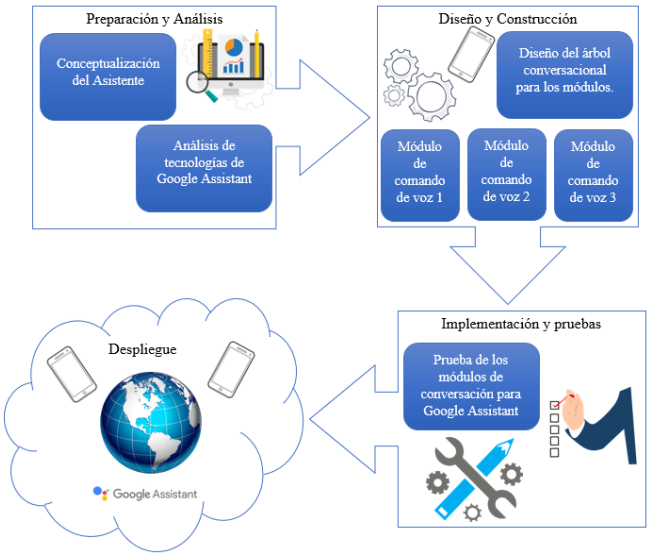
Cheng et al. [12], desarrollaron y evaluaron una aplicación con el asistente de *Google Home* que actúa como una estrategia de intervención innovadora para ayudar a los pacientes ancianos con el autocontrol de la diabetes tipo 2. Este marco de aplicación combinó la interfaz de voz de Google Home para alojar el agente de conversación utilizando la plataforma de API.ia hoy conocido como Dialogflow en conjunto con el SDK de *Google* y una interfaz web para la visualización de datos se creó un *WebHook*, el cual consiste en devolución de llamada HTTP (*Hypertext Transfer Protocol*, Protocolo de Transferencia de Hipertexto) utilizando Node.js como plataforma, este utiliza *JavaScript* con lenguaje de programación y con el fin de reducir la carga de monitoreo sobre las consecuencias diabéticas para el usuario. Se realizó una comparación basada en las características de la aplicación con las aplicaciones móviles disponibles actualmente para determinar el cumplimiento relativo de los componentes oficiales. Como resultado la aplicación mejoró el estado actual de la técnica al aumentar la satisfacción y conveniencia del usuario.

La literatura actual no ofrece suficiente información de casos relacionados con el desarrollo de nuevos comandos e instrucciones para el asistente virtual de *Google*, ya que éste no ha sido completamente explotado, debido a que en algunos casos resulta ser difícil de utilizar o de estudiar.

4. PROCESO DE DESARROLLO

Para cumplir con el objetivo de este trabajo, se utiliza la metodología descrita en la Figura 1 que se muestra a continuación.

Figura 1. Proceso para el alcance de la solución del proyecto



1. Preparación y análisis: En esta etapa, se analizan diferentes investigaciones disponibles en las bibliotecas digitales de ACM *Digital Library*, *Springer Link*, *Science Direct* y *IEEEXplore*, con el fin de encontrar resultados sobre implementaciones de nuevos comandos de voz y qué tecnologías se emplearon en esos casos.

2. Diseño y construcción: En esta fase se creará un árbol conversacional capaz de prever todos los puntos de contacto entre el usuario y el asistente inteligente, así como las respuestas a preguntas mal formuladas o incluso a insultos,

para minimizar los posibles errores durante el proceso de conversación.

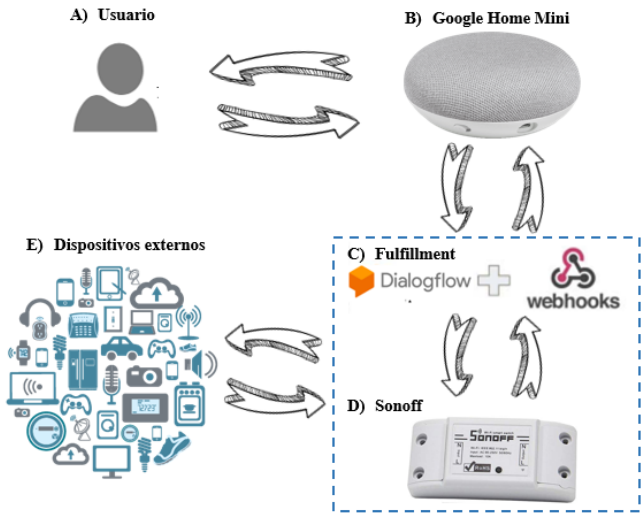
3. Implementación y pruebas: En este paso se realizará la implementación y prueba de los nuevos comandos de voz utilizando las siguientes tecnologías de *Google*: un simulador capaz de emular un entorno de prueba deseado, un dispositivo *Google Home mini* y un *Smartphone*.

4. Despliegue: Corresponde a la publicación de los nuevos comandos de voz en la plataforma de *Google Assistant*, tomando como referencias los hitos publicados para desarrolladores de la comunidad de *Google*.

5. ARQUITECTURA DE DESARROLLO

En la Figura 2, se presenta la arquitectura propuesta en este artículo; es importante destacar, que dicha arquitectura se construyó basándose en la misma que maneja *Google*, por lo que es necesario atender las actualizaciones que la empresa realice en las tecnologías usadas para el desarrollo de las actividades.

Figura 2. Arquitectura para el desarrollo de los nuevos comandos de voz



A) Usuario: corresponde a quien utilizará el lenguaje natural para comunicarse con el dispositivo *Google Home Mini*, es importante resaltar que es necesario usar la frase "Ok Google" antes de cualquier instrucción, para que el dispositivo reconozca cualquier cosa después de dicha oración como una orden.

La frase "OK Google", ya se encuentra definida y es un punto clave para desencadenar la acción deseada; esta frase no se puede cambiar y es de uso exclusivo del *Google Assistant*.

B) Google Home Mini: el dispositivo obtendrá la orden expresada por el usuario, y se encontrará conectado a la plataforma de *Dialogflow*.

El *Google Home Mini* es el dispositivo físico por medio del cual se puede hacer uso del Google Assistant, dado que cuenta con hardware particular que permite una mejor captación de sonido en lugares más amplios, evitando el desfase de error por eco al momento de recibir la orden.

C) Fullfillment: es el servicio encargado de procesar el lenguaje natural de la instrucción que manda el usuario, su objetivo es encontrar en dicha instrucción, las palabras claves que sirven para identificar el comando de voz que se requiere, su acción puede ser complementada con *WebHooks*, como se observó en [8] y [12].

El *Fullfillment* es donde se manejará la mayor parte de la lógica para lograr que el comando de voz pueda hacer la interacción y/o nuevas funcionalidades que se requieran para que este pueda lograr su objetivo.

La interacción con el dispositivo de Google y el Sonoff se llevará a cabo mediante uno o varios JSON, dependiendo de los comandos y/o acciones que se requieran. El JSON es un formato de texto sencillo el cual permite el intercambio de datos de manera más accesible logrando agilizar el proceso de comunicación.

En este JSON se mostrarán las acciones que ejecutará el comando de voz; además, el archivo JSON se apoyará del lenguaje de programación que admita gRPC para especificar funcionalidades extras y tener un buen funcionamiento. En esta sección se encuentra la mayor responsabilidad sobre la arquitectura.

D) Sonoff: es un dispositivo que actuará de intermediario para la conexión con los dispositivos externos; se planea controlar mediante comandos de voz.

Este dispositivo tiene la particularidad de permitir el entendimiento con el asistente virtual de *Google* y ofrecer un mejor control sobre los dispositivos externos como el *Raspberry Pi* [10].

E) Dispositivos externos: estos realizaran una acción dependiendo del comando de voz que se haya ejecutado, por ejemplo, sea el caso de que este dispositivo este representado por un *Smartphone*, una acción válida para tal, será encenderse o apagarse.

6. CONCLUSIONES Y RECOMENDACIONES

Gracias a los trabajos en los que se apoya este artículo se logró comprender el funcionamiento que abarcan los asistentes virtuales, además se adquirió el conocimiento necesario sobre el asistente virtual de *Google*, el cual cuenta con un amplio repertorio de tecnologías útiles para desarrollar nuevos comandos de voz.

Anteriormente la creación de los comandos de voz usando el SDK de *Google Assistant* era mediante su biblioteca nativa en *Python*, árboles conversacionales y el servicio de *Google*, sin embargo, las últimas actualizaciones indican que *Google* abandonó la creación de nuevos comandos de voz por medio de los primeros dos, por lo tanto, los nuevos comandos de voz se deben crear e implementar usando el servicio de *Google Assistant*.

Bajo esta arquitectura se ha desarrollado un agente conversacional que indaga el nombre del interlocutor, pregunta por su color favorito y define un fin en la conversación con diferentes posibles respuestas, dicha conversación es el primer desarrollo que permite comprobar la funcionalidad de la herramienta y de la arquitectura propuesta, dando pie a nuevas aplicaciones posibles.

La arquitectura propuesta para el desarrollo e implementación de un conjunto de instrucciones nuevas para el asistente virtual *Google Assistant*, es una forma idónea de ayudar a los usuarios a automatizar su forma de vida desde el punto de vista domótico, además, dicha arquitectura pretende ser intuitiva para que sea referencia de trabajos futuros.

7. TRABAJO A FUTURO

Se probará la arquitectura propuesta en este trabajo, por lo que se desarrollaran e implementaran nuevos comandos e instrucciones para el asistente virtual de *Google*, para demostrar la utilidad y eficacia de la arquitectura propuesta.

De la misma forma se proponen diversos casos de estudio (*Walkie-talkie*, localizar celular, encender y apagar dispositivos externos) que sirvan como guía para agregar más funcionalidades al asistente virtual de *Google*.

8. AGRADECIMIENTOS

Los autores agradecen el financiamiento del consejo Nacional de Ciencia y Tecnología (Conacyt) y el apoyo del Tecnológico Nacional de México.

9. REFERENCIAS

[1] A. Campos Albuixech, “Asistente virtual en Telegram para acceder a la información económica municipal del Ajuntament de València”, oct. 2018.
[2] S. Junestrand, X. Passaret, y D. Vázquez, Domótica y hogar digital. Editorial Paraninfo, 2004.
[3] E. Méndez y J. A. Moreiro González, “Lenguaje natural e indización automatizada”, Ciencias de la Información, 1999. [En línea]. Disponible en: http://eprints.rclis.org/12685/. [Consultado: 07-abr-2019].
[4] M. B. Hernández y J. M. Gómez, “Aplicaciones de Procesamiento de Lenguaje Natural”, Rev. Politécnica, vol. 32, núm. 0, jul. 2013.
[5] P. Lara Navarra y J. A. Martínez Usero, Agentes inteligentes en la búsqueda y recuperación de información. Planeta UOC, 2004.
[6] Google, “Actions on Google Glossary | Actions on Google”, Google Developers. [En línea]. Disponible en: https://developers.google.com/actions/glossary. [Consultado: 15-mar-2019].
[7] Google, “Google Assistant SDK”, Google Developers. [En línea]. Disponible en: https://developers.google.com/assistant/sdk/overview. [Consultado: 14-mar-2019].
[8] G. Iannizzotto, L. L. Bello, A. Nucita, y G. M. Grasso, “A Vision and Speech Enabled, Customizable, Virtual Assistant for Smart Environments”, en 2018 11th International Conference on Human System Interaction (HSI), 2018, pp. 50–56.
[9] A. H. Michaely, X. Zhang, G. Simko, C. Parada, y P. Aleksic, “Keyword spotting for Google assistant using contextual speech recognition”, en 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2017, pp. 272–278.
[10] C. Peng y R. Chen, “Voice recognition by Google Home and Raspberry Pi for smart socket control”, en 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI), 2018, pp. 324–329.
[11] I. Hwang, J. Jung, J. Kim, Y. Shin, y J. Seol, “Architecture for Automatic Generation of User Interaction Guides with Intelligent Assistant”, en 2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA), 2017, pp. 352–355.
[12] A. Cheng, V. Raghavaraju, J. Kanugo, Y. P. Handrianto, y Y. Shang, “Development and evaluation of a healthy coping voice interface application using the Google home for elderly patients with type 2 diabetes”, en 2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC), 2018, pp. 1–5