

Propuesta de una arquitectura para un Smart Campus Universitario

Proposal of an architecture for a Smart University Campus

CRUZ-PARADA, José María^{†*}, ZAMUDIO-RODRIGUEZ, Víctor Manuel, LINO-RAMÍREZ, Carlos y GUTIERREZ-HERNANDEZ, David Asael

Instituto Tecnológico de León. División de Investigación y Estudios de Postgrado. Leon, Gto. México

ID 1^{er} Autor: *José María, Cruz-Parada* / ORC ID: 0000-0003-0196-1666, CVU CONACYT ID: 803748

ID 1^{er} Coautor: *Víctor Manuel, Zamudio-Rodriguez* / ORC ID: 0000-0002-9246-7999, CVU CONACYT ID: 70912

ID 2^{do} Coautor: *Carlos, Lino-Ramírez* / ORC ID: 0000-0002-6415-8435, CVU CONACYT ID: 395781

ID 3^{er} Coautor: *David Asael, Gutierrez-Hernandez* / ORC ID: 0000-0002-9374-5110, CVU CONACYT ID: 167864

DOI: 10.35429/JSI.2020.14.4.21.27

Recibido Abril 19, 2020; Aceptado Junio 30, 2020

Resumen

Se describe una arquitectura para el uso de agentes inteligentes conectados a una aplicación móvil que a su vez también está enlazado a un sistema de control que es administrado por la institución. En este documento se analiza la idea desde su concepción, pasando por el desarrollo elaborado y las pruebas y los resultados que se han llevado a cabo. Esta arquitectura se planea utilizar en la creación de un campus universitario inteligente con recolección de datos, análisis de información y toma de decisiones automatizadas.

Agentes inteligentes, Arquitectura tecnológica, Campus inteligente, Ambientes inteligentes

Abstract

A proposal of an architecture is described for the use of intelligent agents connected to a mobile application and the same time is also linked to a control system that is managed by the institution. In this document the idea is analyzed from its conception, through the elaborated development and the tests and the results that have been carried out. This architecture is planned to be used in the creation of an intelligent university campus with data collection, information analysis and automated decision making.

Intelligent environments, Smart campus, Technology architecture, Intelligent agents

Citación: CRUZ-PARADA, José María, ZAMUDIO-RODRIGUEZ, Víctor Manuel, LINO-RAMÍREZ, Carlos y GUTIERREZ-HERNANDEZ, David Asael. Propuesta de una arquitectura para un Smart Campus Universitario. Revista de Innovación Sistemática. 2020. 4-14:21-27.

*Correspondencia al Autor (Correo Electrónico: chema@itleon.edu.mx)

† Investigador contribuyendo como primer autor.

Introducción

Los *ambientes inteligentes* han tenido un gran impacto en los últimos años, debido a nuevas herramientas basadas en sistemas embebidos (sensores y actuadores), que facilitan tener sistemas de cómputo de bajo costo, pero con buenas capacidades para comunicar, sensor y actuar a través un enfoque basado en el análisis de datos. Esto ha dado lugar a nuevos paradigmas como el *Internet de las cosas* (IoT) y la *Industria 4.0*. En el entorno académico, estos nuevos paradigmas cobran especial importancia, pues permiten ofrecer servicios basados en las necesidades del usuario, ya sea aprendiendo comportamientos, detectando situaciones atípicas o prediciendo situaciones de riesgo y actuando en consecuencia.

El proyecto denominado como ‘Smart Campus’, es una propuesta creada para integrar agentes que estén alerta del ambiente y puedan tomar decisiones para generar interacción con los usuarios del Tecnológico Nacional de México en León. Para lograr este objetivo, se propone una arquitectura que busca ser un complemento del sistema integral de procesos que ya cuenta la institución. La estructura propuesta busca tener la posibilidad de agregar o quitar agentes ya sea para actualización, mantenimiento, etc.

En este artículo se presentará el estado del arte para la creación de la propuesta, la arquitectura propuesta donde se analiza los componentes que la forma, los resultados alcanzados hasta el momento, el trabajo futuro que se plantea realizar y las conclusiones correspondientes.

Estado del arte

Agente inteligente

Se considera como *agente inteligente* a un sistema que puede realizar un proceso automatizado para alcanzar un objetivo, con la condición de que el sistema reconoce y actúa sobre un entorno, ya que puede percibir información, comunicar o recibir datos y realizar una acción correspondiente (Serna et al., 2019). Es importante señalar que el agente realiza acciones sobre un ambiente al estar situado en el interior del (Altube, Sattolo y Lipera, 2016).

Ambientes inteligentes

Un *ambiente inteligente* es descrito como un espacio que está envuelto en un proceso de digitalización por medio de equipos computacionales que pueden sensor y responder (Nehmer et al., 2006) a ciertos objetivos que ayudan a los usuarios involucrados. Ya que se pueden crear tareas que realizan acciones a través de las decisiones que pueden ser llevados a cabo por personas o sistemas automatizados como *agentes* (Ramos, Augusto y Shapiro, 2008).

Campus inteligentes

La idea del *campus inteligente* se deriva del concepto de ciudad inteligente que aplica el conjunto de información recabada y los procesos relacionados para lograr una administración eficiente de los recursos y los bienes que son utilizadas en las ciudades, ya que se utiliza información, comunicación y elementos que se integran a través de diversos dispositivos conectados alrededor de este entorno (*ambiente inteligente*) (Fortes et al., 2019). Por lo tanto, las instituciones educativas pueden aprovechar estas características al ser un modelo de comunidad que cuenta con actividades parecidas a una ciudad pequeña como administración, gobernanza, sustentabilidad y aprendizaje (Fortes et al., 2019).

Se puede resumir que un *campus inteligente* utiliza soluciones de IoT para conectar, monitorear, controlar, optimizar y automatizar los sistemas de una universidad (Fraga-Lamas et al., 2019). Existen casos donde se han realizado trabajos relacionados al tema como la plataforma basada en IoT implementada en la institución educativa Moncloa Campus de Excelencia Internacional, ubicado en la ciudad de Madrid, España y cuyo objetivo es experimentar con servicios basados en *Ciudades Inteligentes* (Alvarez-Campana, López, Vázquez, Villagrà & Berrocal, 2017). Otro caso es el diseño de una arquitectura basada en computación en la niebla para aplicaciones de campus inteligentes habilitadas para IoT (Fraga-Lamas et al., 2019).

Arquitectura propuesta

Base de la arquitectura

En la realización de la arquitectura, se utilizó como base la infraestructura tecnológica que se encuentra existente en el Tecnológico Nacional de México en León, como se puede observar en la figura 1. En la arquitectura podemos distinguir los diferentes elementos que lo conforman y que se describen en los siguientes párrafos.

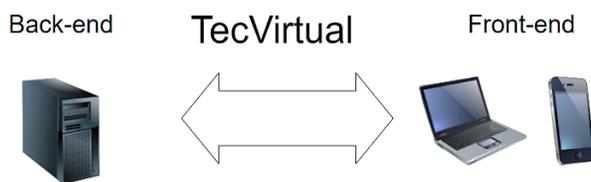


Figura 1 Arquitectura de Sistema Integral del TecNM en León

Fuente: elaboración propia [Draw.io - Diagrams.net]

TecVirtual es un sistema que integra la mayoría de los procesos que se realizan de manera automatizada en el Tecnológico Nacional de México en León, por ejemplo, el proceso de ingreso de nuevos alumnos. Este sistema se divide en dos grandes partes que se manejan en el desarrollo de software, los cuales se explican en los siguientes párrafos.

Por un lado, tenemos el concepto de *Front-end* el cual se dedica a desarrollar principalmente interfaces para mostrar información, con el objetivo de ser visualizado o interpretado por los usuarios. Otro punto es la recolección de datos por parte del usuario y también se puede hacer un control dependiendo un proceso, a través de posibles acciones. Del otro lado se encuentra el *Back-end*, el cual se desarrolla para generar los servicios donde se procesa y se analiza la información, así como el lugar donde se realiza el almacenamiento correspondiente mediante *bases de datos* o *sistemas de almacenamiento*. Un punto a destacar es la *seguridad* como una capa intermedia entre cualquier usuario que haga una petición y los servicios que se ofrecen de manera digital.

Desarrollo de la arquitectura

Ya que se conoce la base de la arquitectura, se establece cuáles son las entidades que van a interactuar en la infraestructura. Uno de ellos son los *agentes*, representados en la figura 2, de los cuales vamos a tener una cantidad indefinida que estarán esparcidos por todo el *ambiente* y el otro actor necesario, con el objetivo de realizar la comunicación entre dispositivos, es un *servidor de sockets*, el cual se puede visualizar en la figura 3.

Agents



Figura 2 Representación gráfica de un agente

Fuente: elaboración propia [Draw.io - Diagrams.net]

Socket



Figura 3 Representación gráfica del servidor de sockets

Fuente: elaboración propia [Draw.io - Diagrams.net]

La decisión trabajar con el manejo de *sockets* es que la comunicación que se haga en tiempo real, como se usa comúnmente en la mensajería instantánea, aparte de que este método de comunicación permite tomar decisiones o tomar acciones al momento de controlar el envío de mensajes, por ejemplo, el poder detectar comportamientos al momento de estar mandando información.

El uso de los *agentes* es indispensable por sus características, el estar alerta de su entorno es lo que se ajusta perfectamente para el proyecto de *campus inteligente*, dado que como está conectado todo el tiempo a una red, se está mandando y recibiendo información con la cual se puede contar con que la comunicación a dicho *agente* es viable.

En el tema de la comunicación por *sockets*, entre *agente*, debe de ser bidireccional, dado que un *agente* puede recolectar datos y mandar un mensaje, también puede recibir o hacer peticiones para recibir información, como se muestra en la figura 4.

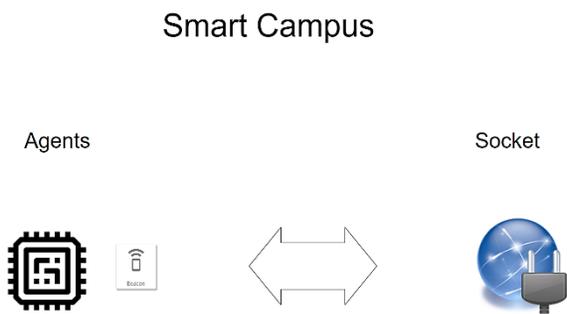


Figura 4 Interacción entre agentes y servidor de sockets
Fuente: elaboración propia [Draw.io - Diagrams.net]

A la interacción de los *agentes* y el *servidor de sockets*, se debe sumar lo que ya se cuenta del sistema integral, el cual ya tiene el objetivo de tener una presentación de la información y control de algunas acciones del lado de los usuarios y para esto se ha implementado aplicaciones con interfaces, tanto en navegador web como en sistemas móviles, que son parte del *Front-end*. Por otro lado, como parte de *Back-end*, se manejan los servicios que se comunican con la seguridad, los servicios para diferentes procesos y servicios de almacenamiento en base de datos, mismos que son representados gráficamente en la figura 1.

Como consecuencia de lo detallado anteriormente, el siguiente paso es conectar esta arquitectura del ambiente inteligente con el sistema integral ya existente, por lo tanto, se propone que los *agentes* van a estar en constante comunicación con los demás dispositivos, sin embargo toda la comunicación a los usuarios va a ser mediante los *sockets*, entonces la comunicación entre *agente* y usuario mediante la aplicación móvil va en una dirección, saliendo del *front-end* rumbo a los agentes pero no en el sentido contrario.

Al mismo tiempo todos deben estar conectados con los *sockets*, dado que es la forma en la cual podemos centralizar toda la información que viaja en el sistema, con esta información se llega a tener una arquitectura definida y que se muestra en la figura 5.

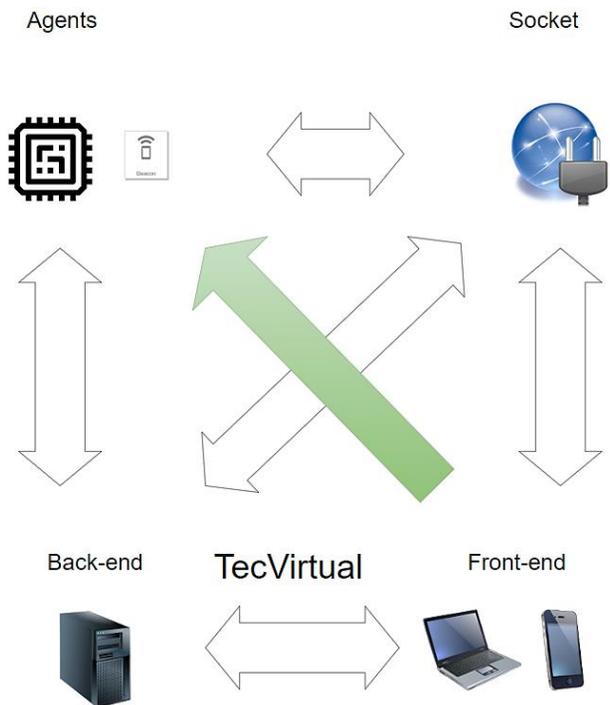


Figura 5 Propuesta de arquitectura para el campus inteligente
Fuente: elaboración propia [Draw.io - Diagrams.net]

Construcción de la arquitectura

Del lado del software, un *agente* está construido mediante el lenguaje de programación de *Python*, mientras del lado de hardware, está funcionando un ordenador conocido como *Raspberry Pi 3*, que se visualiza en figura 6.

El uso de este ordenador de placa única es debido a que tiene integrado componentes que realizan la comunicación mediante tecnologías *Wi-Fi* y *Bluetooth*, las cuales se necesitarán en el proyecto, ya que un *agente* contendrá distintos procesos que generan la capacidad de tomar decisiones conforme cambie su entorno y esto puede ser posible utilizando los componentes que se manejan en el hardware.

Regresando al tema de la programación, se utilizará la biblioteca de programación PADE (Python Agent DEvelopment framework), que es una herramienta que permite realizar la ejecución de agentes, en este caso, se ha realizado una prueba, donde se genera el agente y se manda un mensaje a la consola de comandos, para demostrar que ha funcionado correctamente, esto se puede visualizar en la figura 7.



Figura 6 Ordenador de placa Raspberry Pi 3
Fuente: elaboración propia [Fotografía]

```
connection established
[Agente_1] 29/02/2020 14:41:31.168 --> ¡Estoy escuchando!
[Agente_1] 29/02/2020 14:41:32.169 --> ¡Estoy escuchando!
[Agente_1] 29/02/2020 14:41:33.170 --> ¡Estoy escuchando!
```

Figura 7 Primeras pruebas de un agente
Fuente: elaboración propia [Linux Terminal]

Continuando con la elaboración de pruebas, se desarrolló una aplicación móvil desde cero, con el objetivo de comprobar la comunicación entre un sistema cliente-servidor mediante el uso de *sockets* poniendo de prueba la función de un chat, el desarrollo de la aplicación está implementado mediante la tecnología Ionic.

El beneficio de utilizar dicha herramienta impacta directamente en el tiempo de desarrollo y dinero que se invierte en cualquier aplicación, dado que solo se tiene que trabajar en un desarrollo híbrido, de lo contrario se tendría que desarrollar una aplicación por sistema operativo (Android y iOS) para dispositivos móviles o de escritorio y web. Como consecuencia del uso de Ionic, se realizó la aplicación con un diseño muy sencillo, tipo chat, con el objetivo de llevar a cabo la prueba en un teléfono inteligente con sistema operativo Android, como se muestra en la figura 8.

Chat Room

Enter message

Figura 8 Aplicación móvil para pruebas
Fuente: elaboración propia [Captura de pantalla en dispositivo móvil]

El *servidor de sockets* se desarrolla usando el lenguaje de programación *NodeJS* y la biblioteca *socket.io*, la cual cuenta con una gran cantidad de apoyo por medio de la comunidad de desarrollo y una documentación abundante para su futura implementación y crecimiento. A través del servidor, se permite controlar la comunicación entre usuarios, de agentes a usuarios o entre agentes, al mismo tiempo que se podría tomar decisiones con la información.

Resultados

Para ver los resultados, se realizó una prueba en la aplicación móvil, donde se planteó mandar un mensaje entre distintos dispositivos para revisar la comunicación entre ellos, el primer dispositivo escribe “Aplicación 1” y el segundo escribe “Aplicación 2”, esto debe verse reflejado de la siguiente manera, los mensajes en color azul, son los mensajes que se escribieron y se mandaron desde un dispositivo cliente, mientras los mensajes en color gris, son los que se recibieron a través del servidor de sockets y se reenviaron al otro dispositivo, los resultados de la prueba se muestran en la figura 9.

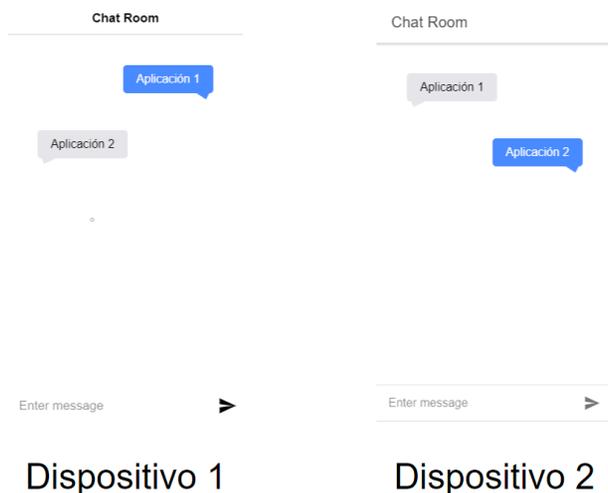


Figura 9 Pruebas con aplicación móvil y sockets
Fuente: elaboración propia [Draw.io - Diagrams.net]

Después se manda un mensaje a sólo un dispositivo mediante un agente y al mismo tiempo se manda otro mensaje a todos los dispositivos, se plantea, como ejemplo, cuando un agente toma una decisión y le manda una recomendación a un usuario específico o a los usuarios en común, todo tiene que pasar por medio de nuestro *servidor de sockets* como se demuestra en la figura 10. Esto se lleva a cabo a través del uso de identificadores para que cada usuario sea tomado como un elemento diferente y saber con quién se está comunicando.

Para tener un identificador, se aprovecha que la aplicación actual maneja como identificador único el CURP (Clave Única de Registro de Población) de los usuarios, por lo tanto, se busca utilizar identificadores de CURP ficticios en la etapa de desarrollo y luego implementarlo a los usuarios que pertenecen al instituto.

Actualmente la Institución cuenta con una matrícula que asciende aproximadamente a los 4500 estudiantes, tomando ese número como base para la prueba, se logró mantener 9000 conexiones estables concurrente, sin impedir el correcto funcionamiento del sistema, teniendo en cuenta que dicho límite de conexiones va de la mano con la capacidad de procesamiento y la cantidad de memoria de nuestros servidores.

Trabajo futuro

Como continuación para el proyecto, se necesita diseñar la arquitectura con la cual se van a conectar nuestros *agentes inteligentes*, que se encuentren en diversas ubicaciones, a un sistema central, en donde los agentes puedan recopilar información al mismo tiempo y toda la información se dirija a un centro de datos que pueda ser usada para diferentes procesos como análisis o métodos de inteligencia artificial, como el manejo de *sistemas difusos*, los cuales ayudan a la toma de decisiones en los diferentes ámbitos que podría atacar el ‘Smart Campus’.

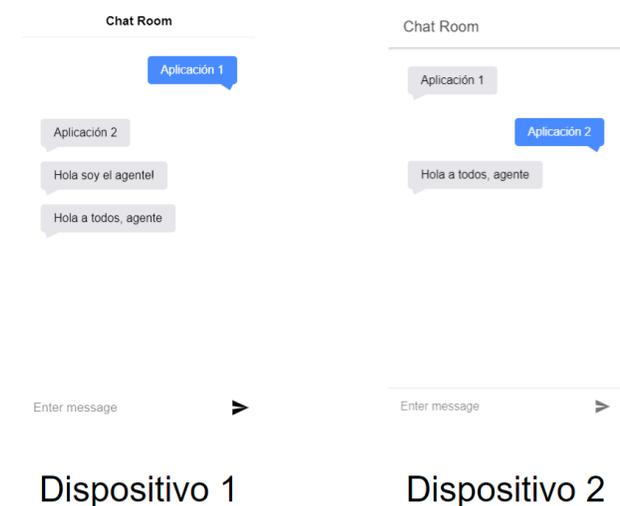


Figura 10 Pruebas de aplicación móvil e interacción con agente
Fuente: elaboración propia [Draw.io - Diagrams.net]

Conclusiones

La propuesta que se describe en el documento, es una idea con base a la necesidad de tener sistemas intercomunicados, en este caso los sistemas con lo que maneja el Tecnológico Nacional de México en León y al mismo tiempo implementar un *campus inteligente*, con el fin de asegurar que esta comunicación funciona correctamente, gracias a la experimentación que se realizó, se puede concluir que es una base sólida la que se está construyendo para poder continuar con este proyecto incrementando el tamaño de *agentes* y usuarios que se pueden encontrar en el entorno.

Agradecimientos

Nos gustaría agradecer al Tecnológico Nacional de México / Instituto Tecnológico de León por el apoyo brindado a esta investigación.

Referencias

Altube, Alejandro, Iris Sattolo y Liliana Lipera (2016). «Bot guía para un ambiente virtual inmersivo». En: XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016).

Alvarez-Campana, M., López, G., Vázquez, E., Villagrà, V., & Berrocal, J. (2017). Smart CEI Moncloa: An IoT-based Platform for People Flow and Environmental Monitoring on a Smart University Campus. *Sensors*, 17(12), 2856.

Fraga-Lamas, P., Celaya-Echarri, M., Lopez-Iturri, P., Castedo, L., Azpilicueta, L., & Aguirre, E. et al. (2019). Design and Experimental Validation of a LoRaWAN Fog Computing Based Architecture for IoT Enabled Smart Campus Applications. *Sensors*, 19(15), 3287.

Fortes, S., Santoyo-Ramón, J., Palacios, D., Baena, E., Mora-García, R., & Medina, M. et al. (2019). The Campus as a Smart City: University of Málaga Environmental, Learning, and Research Approaches. *Sensors*, 19(6), 1349.

Nehmer, Jürgen y col. (2006). Living assistance systems: an ambient intelligence approach. En: Proceedings of the 28th international conference on Software engineering. ACM, pp. 43-50.

Ramos, Carlos, Juan Carlos Augusto y Daniel Shapiro (2008). Ambient Intelligence—the Next Step for Artificial Intelligence». En: *IEEE Intelligent Systems* 23.2, pp. 15—18.

Serna, B., Baltazar, R., Cruz-Parada, P., Meza, J., Manríquez, J. and Zamudio, V. (2019). Use of Intelligent Agent Through Low-Cost Brain-Computer Interface to Analyze Attention and Meditation Levels by Gender. *Agents and Multi-agent Systems: Technologies and Applications 2019*, pp.163-174.