

Utilización de inteligencia artificial y servicios cognitivos en el desarrollo de un prototipo de solución para la búsqueda de medicamentos

RESUMEN: En la actualidad se ha notado un incremento en el precio de los medicamentos en México, a raíz de esto los consumidores se han visto en la necesidad de comparar precios en diferentes farmacias, para disminuir esta situación se propone una solución informática que está compuesta de una aplicación móvil (iOS y Android), una plataforma Web y un Chatbot (Skype y Telegram) que utiliza servicios Cognitivos de Microsoft Azure y LUIS (Language Understanding Intelligent Service) como lenguaje para Inteligencia Artificial e implementarla en la nube de computo de Microsoft Azure, todo esto construido mediante Visual C# con una arquitectura en capas y técnicas como SOLID, Clean Code facilitando el mantenimiento y reutilización del código. En esta se ingresa el nombre del medicamento, muestra las farmacias más cercanas, y una lista de estas, donde usuarios anteriormente han registrado la compra de dicho medicamento, al seleccionar la farmacia deseada mostrará la ruta para llegar a esta.

PALABRAS CLAVE: ChatBot, Cloud Computing, LUIS, Servicios Cognitivos, Inteligencia Artificial, Xamarin.Forms, SOLID, Clean Code, Visual C#.



Colaboración

Espinoza Galicia Carlos Arturo; Ángeles Hernández Anahí; Mendoza Trejo Juan José; Morales Ramírez Esmeralda, Instituto Tecnológico Superior de Huichapan

ABSTRACT: Currently, there has been an increase in the price of medicines in Mexico, as a result of which consumers have had to compare prices in different pharmacies, in order to reduce this situation, a computer solution is proposed that is composed of a mobile application (iOS and Android), a Web platform and a Chatbot (Skype and Telegram) that uses Microsoft Azure and LUIS Cognitive Services (Language Understanding Intelligent Service) as a language for Artificial Intelligence and implement it in the Microsoft Azure computing cloud, all this is built using Visual C# with a layered architecture and techniques such as SOLID, Clean Code facilitating the maintenance and reuse of the code. In this the name of the medicine is entered, it shows the closest pharmacies, and a list of these, where users have previously registered the purchase of said medication, when selecting the desired pharmacy will show the route to reach it.

KEYWORDS: ChatBot, Cloud Computing, LUIS, Cognitive Services, Artificial Intelligence, Xamarin.Forms, SOLID, Clean Code, Visual C#.

INTRODUCCIÓN

De acuerdo a varias fuentes de información el precio de los medicamentos se ha visto afectado, incrementando su valor comercial que va del 3 al 5 por ciento aproximadamente, mientras que en el Estado de México algunos antibióticos subieron en un 30 por ciento hacia el mes de marzo del presente año, impactando de manera negativa la economía de la población dentro del país [1]

Gran parte de la población se preocupa por su economía sin dejar de lado la calidad de lo que consume, es por ello que se dan a la tarea de realizar una comparativa en el precio y marca de las diferentes farmacias, trasladándose de una a otra para así poder determinar cuál es su mejor opción y realizar finalmente su compra; Dado que este proceso es tedioso y repetitivo, surge la necesidad de crear una solución informática que permita a los usuarios realizar la búsqueda de algún medicamento desde su teléfono celular de manera más rápida y fácil.

La solución mencionada permitirá al usuario realizar búsquedas del medicamento mediante el nombre genérico, comercial, y en casos más específicos por medio de la marca, la búsqueda será realizada por medio de una base de datos que será alimentada por los mismos usuarios respecto a sus experiencias de compra; dichas búsquedas se harán entre las farmacias cercanas al usuario, mostrando detalles de cada establecimiento (nombre del lugar y dirección), el medicamento buscado (Nombre(s) y precio), permitiendo al usuario seleccionar la farmacia que contenga el medicamento según las especificaciones proporcionadas por el mismo, una vez seleccionado el producto a comprar y en caso de requerirlo, el usuario podrá visualizar la ruta para llegar al lugar que ha seleccionado.

De forma general la solución consiste de una aplicación móvil multiplataforma (disponible para Android e iOS), una plataforma web y un Chatbot disponible para Skype y Telegram. En el desarrollo de este proyecto se hace uso de servicios cognitivos de Microsoft Azure y LUIS (Language Understanding Intelligent Service) para el procesamiento de lenguaje natural del ChatBot, Visual C# como lenguaje de programación, y programación en capas para optimizar el tiempo y código del proyecto, entre otras técnicas que serán detalladas a continuación.

MATERIAL Y MÉTODOS

Para el desarrollo del proyecto, se siguen dos metodologías: SCRUM para el seguimiento de proyectos [2] y desarrollo en capas y buenas prácticas para la construcción del software [3] [4].

Las metodologías ágiles están basadas en cuatro valores y permiten desarrollar proyectos de forma más acertada con las necesidades del cliente y responder mejor a los cambios que pudieran surgir al largo del proyecto. Pretenden ofrecer una alternativa más efectiva que los procesos tradicionales caracterizador por la rigidez y dominados por la documentación.

Los valores que rigen esta metodología son [5]:

1. Valorar a las personas y las interacciones entre ellas por sobre los procesos y las herramientas.
2. Valorar el software (producto) funcionando sobre la documentación detallada.
3. Valorar la colaboración con el cliente por sobre la negociación de contratos.
4. Valorar la respuesta a los cambios por sobre el seguimiento estricto de los planes.

En Scrum un proyecto se ejecuta en ciclos temporales cortos y de duración fija (iteraciones que normalmente son de 2 semanas, aunque en algunos equipos son de 3 y hasta 4 semanas, límite máximo de feedback y reflexión). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite. Un ejemplo de esto se encuentra en la figura 1 donde se explica brevemente el proceso a seguir [6].

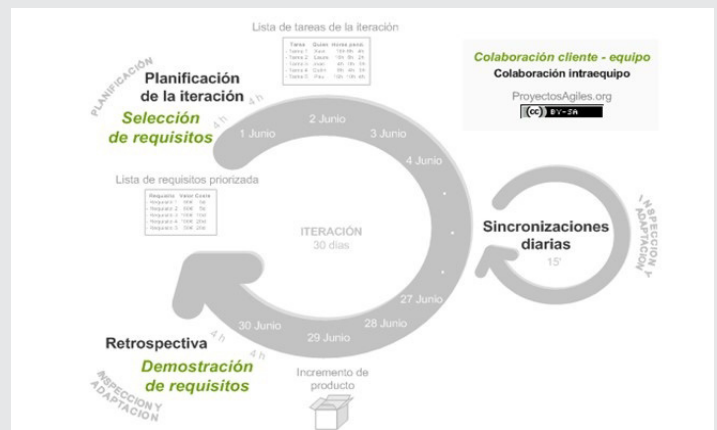


Figura 1 Metodología de seguimiento de proyectos SCRUM. [6]

Para el desarrollo técnico de la solución se utilizó el lenguaje de programación Visual C# [7], como se puede apreciar en la figura 2, se utilizó un modelo de capas [8] utilizando buenas prácticas de programación y principios como Clean Code [9], KISS [10] y SOLID [11], con el propósito de mantener el código entendible, ordenado, abierto para futuras modificaciones y actualizaciones y permitiendo la inclusión de la capa base en los proyectos de la interfaz gráfica. Las capas implementadas se describen a continuación:

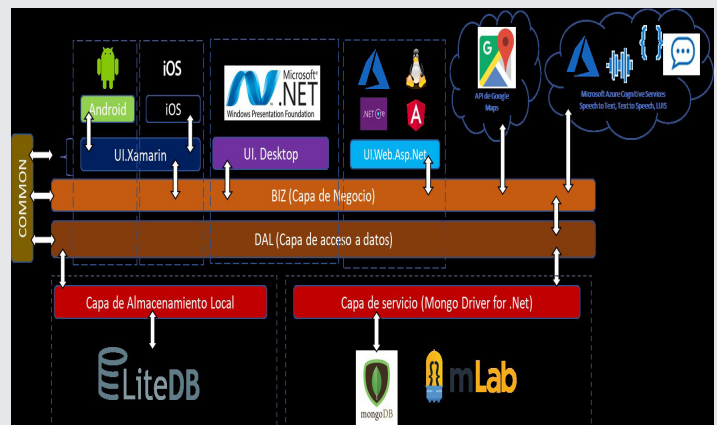


Figura 2 Estructura en capas del proyecto. Fuente: Creación Propia

- Capa Común (COMMON): Se encarga de mapear todas las entidades que se ocupan en el desarrollo de todo el software y es la única que puede ser instanciada en todas las demás capas.

- Capa de Acceso de Datos (DAL): Cuenta con métodos CRUD (Create, Read, Update, Delete) [8] y se comunica con el servidor de MLab en el cual se hospeda la base de datos de MongoDB, permitiendo el transporte de datos(entrada o salida); esta capa no mantiene comunicación directa con el usuario, solamente se comunica con la capa de negocios (BIZ) y utiliza la capa común(COMMON) para obtener las entidades.

- Capa de Lógica de Negocios (BIZ): Contiene la lógica que realiza las funciones principales de la solución que se comparte entre las aplicaciones móvil, web y escritorio. Está directamente en relación con la capa DAL, en esta se encuentran todos los métodos que permiten hacer cualquier proceso con la información ingresada por el usuario y/o extraída de la base de datos y mostrada en la Interfaz. Aquí se encuentran los manager [8] de cada objeto que se formaron implementando la interfaz del manager genérico ubicada en la capa COMMON, teniendo consigo, aparte de los métodos CRUD, sus métodos individuales.

- Capa de Interfaz de Usuario (UI): Se divide en diferentes proyectos con utilidad para los usuarios y administradores. Para cada plataforma se podrá acceder únicamente con usuario y contraseña evitando que cualquier usuario tenga acceso.

- Aplicación Escritorio (WPF): La interfaz de escritorio de Windows se desarrolló usando WPF. Esta aplicación es para uso exclusivo de los administradores de la solución, en los cuales los propósitos principales son: la visualización, edición, inclusión y eliminación de información que es registrada por los usuarios y los administradores ya que se tendrá acceso a la misma base de datos de la aplicación móvil.

- Aplicación Móvil (Multiplataforma): Para el desarrollo de la parte Móvil fue utilizado Xamarin.Forms [12] con el propósito de poder desarrollar tanto en iOS como en Android de manera nativa y con un solo proyecto; esta aplicación es utilizada por los consumidores que desean hacer la búsqueda de un medicamento, implementa métodos de Navegación Jerárquica.

- Plataforma Web: Dicha plataforma está construida con ASP.Net en MVC [13] [14] y nos permite visualizar una lista de todos los registros de medicamentos existentes mostrando en la parte superior una barra de búsqueda donde a partir de un modelo creado permite mostrar una lista de resultados y permite obtener la cadena de la cual se hace búsqueda.

- ChatBot: El ChatBot [13] implementa como lenguaje de inteligencia artificial LUIS [15], uno de los servicios cognitivos de Microsoft Azure [14], de esta manera el usuario puede tener una conversación con respuestas

personalizadas y recibir la información del medicamento que necesite por medio de mensajes.

Para contextualizar la implementación de esta solución tecnológica, está pensada para zonas urbanas y/o rurales, porque de ambas partes una población considerada busca economizar, más sin en cambio se puede considerar una limitante que muchas veces en diferentes zonas no es posible la conexión a internet, y por la arquitectura de la solución se necesita de dicha conexión para acceder a la base de datos y obtener los resultados de búsqueda.

RESULTADOS

Como resultados, se cuenta con una solución en Visual Studio 2017 con la estructura mostrada en la figura 3, donde se aprecian los proyectos de las capas base, la aplicación de escritorio, la aplicación web, la aplicación móvil (con sus distintas plataformas) y el Chat Bot.

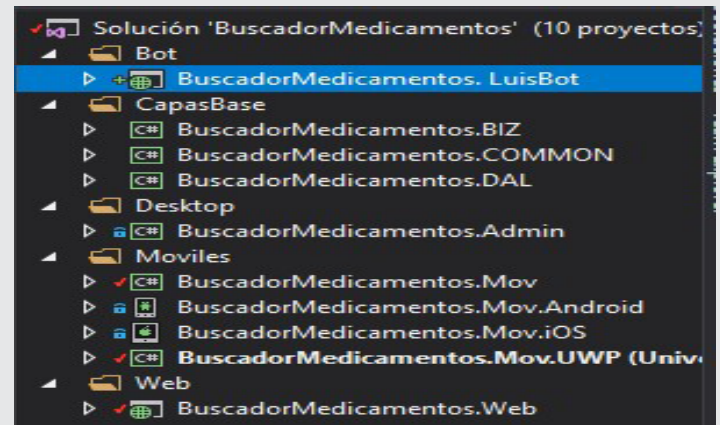


Figura 3 Estructura de la solución en Visual Studio.

Fuente: Construcción Propia.

Hablando de cada proyecto, en la figura 4 se muestra la aplicación de escritorio WPF, cuya finalidad es llevar un control entre los usuarios, medicamentos y farmacias disponibles por los administradores de la solución..

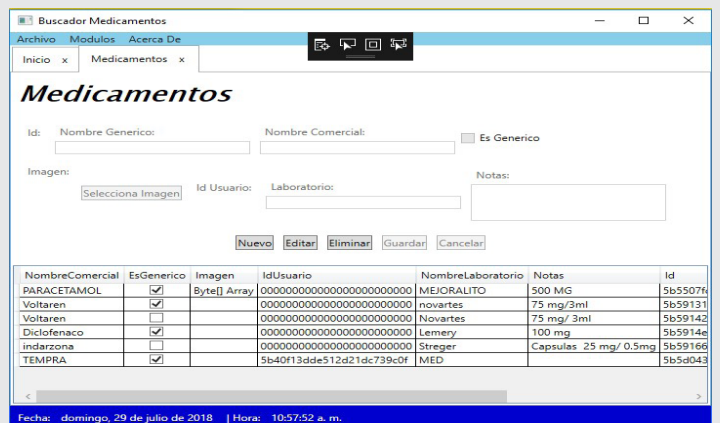


Figura 4 Aplicación de escritorio.

Fuente: Construcción Propia

La aplicación móvil (Android, iOS y UWP para Windows 10), en la figura 5, cuyo funcionamiento está dado mediante Navegación entre Páginas visualizando los medicamentos disponibles y dándote la opción de agregar o editar algún establecimiento y/o medicamento que no esté dado de alta.

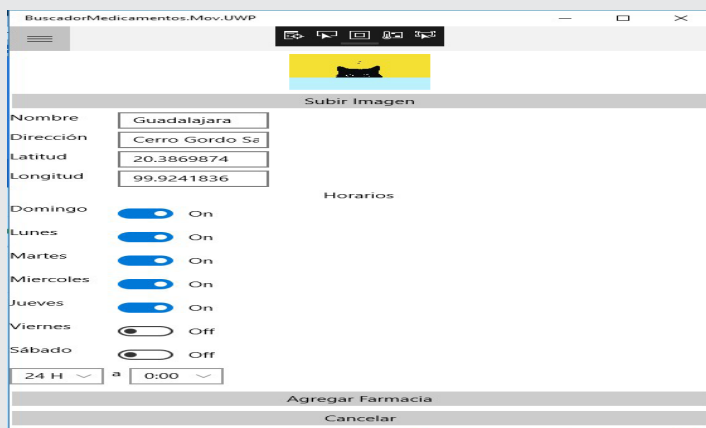


Figura 5. Aplicación móvil versión para Windows 10.

La aplicación web, en la figura 6, construida con ASP. Net donde se utiliza MVC (Model-View-Controller) nos permite visualizar una lista de todos los registros de medicamentos existentes, pudiendo realizar búsquedas específicas mediante una barra de búsqueda, así mismo agregar un medicamento o farmacia, o si es el caso, actualizar sus datos, también crear una cuenta nueva.

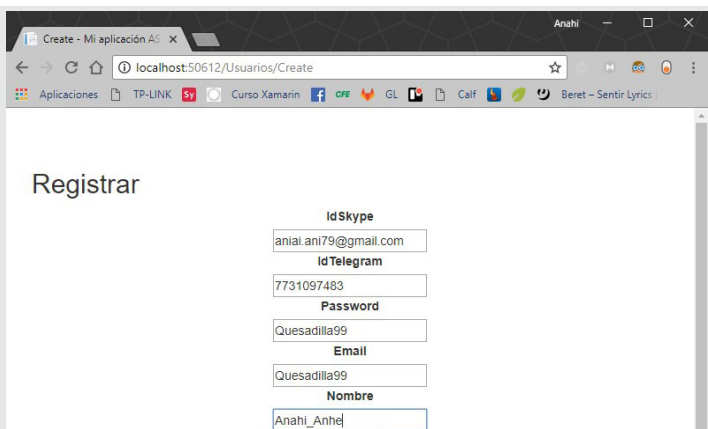


Figura 6 Captura de pantalla la aplicación web, donde se puede crear una cuenta.

Fuente: Construcción Propia

En la figura 7 se muestra una captura de pantalla de una conversación interactiva en Telegram y Skype (mediante el emulador de Microsoft Bot Framework) donde se puede obtener información acerca del precio de un medicamento y la ubicación de las farmacias que lo venden, cabe mencionar que justo es este punto donde se utiliza la parte de Cognitives Services, ya que LUIS se encarga de interpretar el texto recibido en la conversión y entrega intenciones (Pathak, 2017)

las cuales permite identificar el medicamento que está buscando el usuario.

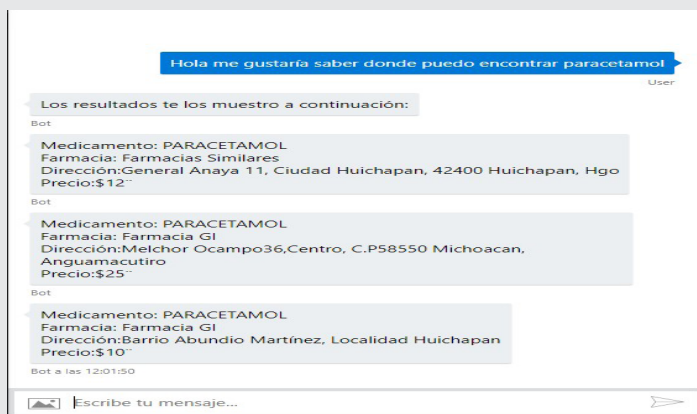


Figura 7 Captura de pantalla del Emulador de Microsoft Bot Framework de la búsqueda de un medicamento.

Fuente: Construcción Propia.

Trabajos Futuros

Para versiones posteriores dentro de este proyecto se planea implementar diferentes aspectos, los cuales son mostrados a continuación:

- Detallar minuciosamente cada plataforma con el fin de obtener una interfaz agradable a la vista, así mismo solucionar errores que posiblemente no se tomaron en cuenta en su primera etapa.
- Realizar pruebas con usuarios potenciales para identificar problemas en el uso de los Servicios Cognitivos y evaluar el funcionamiento de LUIS, para posteriormente realizar mejoras de este.
- Publicar el chat bot tanto en Skype, Telegram y Facebook Messenger.
- Realizar una descarga de manera local al teléfono de los datos de la aplicación, y no sea necesaria siempre la conexión a internet para que el usuario pueda realizar sus búsquedas.
- Lograr que los objetivos que están en espera de una segunda fase se concluyan, es el caso de "Permitir al usuario seleccionar un destino (farmacia) y lo guie hasta la ubicación de esta", se planea utilizar Microsoft Azure Maps o bien la Api de Google Maps para lograrlo, entre otras herramientas.

- Publicación de la plataforma web y las aplicaciones móviles para obtener retroalimentación de usuarios reales.

CONCLUSIONES

Hasta el momento el proyecto es funcional, se continúa con la fase de depuración y mejoramiento de interfaz gráfica de la aplicación de escritorio, aplicaciones móviles, la plataforma web y se evalúa la posibilidad de trabajar con imágenes en el chat bot.

Dicha solución fue elaborada en un solo proyecto haciendo uso de programación en capas permitiendo la reutilización de código y evitando el trabajo que implica el desarrollo de un mismo proyecto en diferente plataforma y sistema operativo.

El proyecto por sí mismo ofrece a los usuarios la comodidad de consultar información de productos desde un mismo sitio y en sus versiones posteriores se planea ser utilizada como una guía para llegar a la farmacia deseada.

Es importante destacar que se hizo uso de inteligencia artificial utilizando los servicios cognitivos que nos brinda Microsoft Azure implementando Language Understanding (LUIS) y un Chat Bot.

BIBLIOGRAFÍA

[1] López Dóriga, «En el último año, se dispararon los precios de las medicinas,» 27 Abril 2018. [En línea]. Available: <https://lopezdoriga.com/economia-y-finanzas/en-el-ultimo-ano-se-dispararon-los-precios-de-las-medicinas/>.

[2] Scrum Manager , Scrum Manager v.2.6, 2016.

[3] M. J. Price, C# 7.1 and .NET Core 2.0 – Modern Cross-Platform Development, Birmingham, UK.: Packt Publishing, 2017.

[4] S. Ritchie, Pro .Net Best Practices, New York: Apress, 2011.

[5] Scrum Mexico, «Scrum & Agile,» 2017. [En línea]. Available: <http://scrum.org.mx/scrum-agile/>.

[6] Proyectos Agiles.com, «¿Qué es Scrum?,» 2016. [En línea]. Available: <https://proyectosagiles.org/que-es-scrum/>.

[7] M. Posadas, Mastering C# and .NET Framework, Birmingham: Packt Publishing Ltd., 2016.

[8] D. Esposito y A. Saltarello, Microsoft .Net: Architecting Applications for the Enterprise, Whashington: Microsoft Press, 2015.

[9] M. Robert C., Clean Code, A Handbook of Agile Software Craftsmanship, Boston, MA: Pearson Education, Inc., 2009.

[10] S. Rossel, «KISS – One Best Practice to Rule Them All,» 19 Agosto 2015. [En línea]. Available: <https://simpleprogrammer.com/2015/08/19/kiss-one-best-practice-to-rule-them-all/>.

[11] A. Gaurav Kumar, SOLID Principles Succinctly, Morrisville, NC: Syncfusion Inc., 2016.

[12] C. Petzold, *Creating Mobile Apps with Xamarin.Forms*, Redmond, Washington, US.: Microsoft Press, 2016.

[13] A. Freeman, *Essential Angular for ASP.NET Core MVC*, Longon, UK.: Apress, 2017.

[14] Microsoft Corporation, «Introducción a ASP.NET Core,» Microsoft, 28 Febrero 2018. [En línea]. Available: <https://docs.microsoft.com/es-es/aspnet/core/?view=aspnetcore-2.1>. [Último acceso: 5 Julio 2018].

[15] N. Pathak, *Artificial Intelligence for .NET: Speech, Language, and Search*, Kotdwara, Dist. Pauri Garhwal, India: Apress, 2017.

[16] Microsoft, «Cognitive Services,» 2018. [En línea]. Available: <https://azure.microsoft.com/en-us/services/cognitive-services/>.

[17] E. Freitas, *Microsoft Bot Framework*, Morrisville, NC, USA: Syncfusion, Inc., 2017.