

Herramienta interactiva en línea como instrumento para el aprendizaje de las matemáticas mediante las técnicas de programación, dirigido a alumnos de educación secundaria

Wilver Aucahuasi, Daniel Vega, Grisi Bernardo, Madelaine Bernardo, Roxana Urbano, Elizabeth Oré
Universidad Continental, Instituto Peruano de Investigación en Ingeniería Avanzada

ABSTRACT – la educación secundaria es el pilar en el desarrollo de habilidades y competencias de las ciencias básicas, permitiendo desarrollar en el alumno habilidades que le permitirán elegir una carrera profesional, el curso de matemáticas es el indicado en permitir el desarrollo de habilidades lógicas, analíticas, interpretación y de cálculo, permitiendo desenvolverse con mayor comodidad en los estudios superiores. Las carreras universitarias y técnicas relacionadas a rubros de la ingeniería, tienen un fuerte componente en la matemáticas y el desarrollo de ciertos cursos relacionados al uso de las herramientas computacionales con mayor énfasis en los cursos de programación, son los cursos de programación los que permiten la continuidad del estudiante, en el presente trabajo de investigación, presentamos una herramienta interactiva mediante el cual se presenta al alumno de educación secundaria, temas relacionados con las matemáticas y la manera como se pueden dar solución mediante el uso de las técnicas de programación, desarrollando conjuntamente habilidades matemáticas, lógicas y de programación, permitiendo al alumno mejorar sus habilidades cognitivas y su capacidad en el uso de las diferentes herramientas computacionales, la prueba piloto se realizó en base de 15 alumnos del segundo grado de educación secundaria de un colegio de Lima, donde se le dictó el curso básico de programación usando la herramienta interactiva, elaborando guías de desarrollo con problemas de razonamiento matemático, lo cual permitió analizar la mejora en la manera de enfocar los problemas matemáticos desde una lógica de programación así como la manera del aprendizaje de un lenguaje de programación, en nuestro caso el lenguaje utilizado fue el Python con una herramienta interactiva basada en inteligencia artificial, que le permite al alumno preguntar en el uso de la sintaxis del lenguaje.

Keywords Python; Procesamiento de lenguaje natural; Agentes inteligentes; Chatbot.

I. INTRODUCCION

En los últimos años la inteligencia artificial y la computación cuántica se han desarrollado muy rápidamente debido en primer lugar a la necesidad de “identificar una explicación científica que diera cuenta en una actividad totalmente ordinaria de cuáles son las operaciones de cálculo que debemos realizar a diario.” (Annovi, 2016)

De esta forma desde la década del 50 del s. XX se han ido gestando diversos modelos teóricos y matemáticos

(máquinas de Turing, redes neuronales, teoría general de sistemas, etc.).

Gracias al incremento en la producción de sofisticados elementos electrónicos es que se ha podido llevar muchos de estos modelos al mundo real siendo uno de los de mayor interés el computador cuántico, por su aplicación a los mercados financieros y sobre todo a la criptografía.

Uno de los lenguajes más favoritos y que está siendo muy aplicado en la inteligencia artificial y la computación cuántica por su fácil curva de aprendizaje y su gran potencia es Python, el cual cuenta con múltiples librerías como numpy y scipy para procesamiento científico de datos, scikit-learn y TensorFlow para inteligencia artificial y QISKit para computación cuántica, las cuales son constantemente usadas, revisadas y mejoradas por una comunidad grande de programadores.

El manejo de los lenguajes de programación requiere de un conocimiento previo en matemáticas y lógica, en las universidades e institutos los cursos de programación cuentan con un alto grado de desaprobados, lo que ocasiona la deserción educativa. la presente propuesta consiste en diseñar una herramienta en línea con la finalidad que sea de ayuda a alumnos de nivel secundario de edades entre los 11 y 15 años, que puedan desarrollar sus cursos de matemáticas usando la herramienta presentada con la intención de lograr un doble desafío, aprender las matemáticas acompañado de aprender técnicas de programación, aumentando la capacidad lógica y percepción del mundo real, necesarios para poder resolver problemas desde el ámbito de la programación.

En el presente trabajo se presenta una herramienta en línea que ayude a entender las estructuras más utilizadas en programación y sus posibles errores que se puedan dar en el momento de ejecución, la herramienta le presenta la forma correcta de escribir el código logrando un acompañamiento en el momento de practicar la escritura y ejecución del código en Python. Con la intención de orientar al estudiante como si fuera el profesor, dando la sensación que tener al profesor presente al momento de realizar la práctica.

Digital Object Identifier (DOI): <http://dx.doi.org/10.18687/LACCEI2018.1.1.573>
ISBN: 978-0-9993443-1-6
ISSN: 2414-6390

En la actualidad, la educación escolar hace uso de las metodologías tradicionales en la enseñanza de los cursos que corresponden a las asignaturas de las matemáticas y computación, estas metodologías utilizan como componentes educativos a los estudiantes, a los profesores y los padres en ese orden de prioridad, lo que ocasiona en la mayoría de los casos un lento aprendizaje de las asignaturas, el avance tecnológico hace posible la germinación de nuevos conceptos y técnicas que hacen posible comprender otros aprendizaje hasta ahora no abarcados como la actividad cerebral en el momento que se desarrollan estas actividades y en tareas relacionadas a la actividad diaria, estos nuevos conceptos están en la línea de la neuroeducación, que esta soportada por el concepto de la enseñanza basada en la actividad cerebral, donde influyen las emociones y los niveles de concentración con ello se pueden diseñar nuevas estrategias y metodologías para el mejor aprendizaje de las matemáticas y la computación en los estudiantes de educación escolar.

Nuestra propuesta está centrada en educación secundaria y se basa en el aprendizaje de las matemáticas con la ayuda de la computación mediante las técnicas de programación, con el objetivo de mejorar el aprendizaje usando los conceptos de la neuroeducación, trabajando con la concentración y optimizando los resultados mediante la interacción con la herramienta interactiva que hace posible ayudar a resolver problemas matemáticos clásicos que se desarrollan en clase con el uso de las técnicas de programación, logrando resultados óptimos, por la comprensión de la lógica de programación y su aplicación en resolver problemas comunes de clase, con el uso de la herramienta tecnológica se aprovechamos el interés que despierta esta nueva forma de abordar y resolver problemas matemáticos, pudiendo escalar hacia la solución de problemas más complejos de análisis matemático, física, diseño de videojuegos entre otros.

II. MATERIALES Y MÉTODOS

A continuación, se define brevemente los componentes de la herramienta:

a) SISTEMA EXPERTO – INTELIGENCIA ARTIFICIAL

Un sistema experto puede definirse como un sistema informático (hardware y software) que simula a los expertos humanos en un área de especialización dada. Como tal, un sistema experto debería ser capaz de procesar y memorizar información, aprender y razonar en situaciones deterministas e inciertas, comunicar con los hombres y/u otros sistemas expertos, tomar decisiones apropiadas, y explicar por qué se han tomado tales decisiones. Se puede pensar también en un sistema experto como un consultor que puede suministrar ayuda a (o en algunos casos sustituir completamente) los expertos

humanos con un grado razonable de fiabilidad. (Castillo, Gutiérrez y Hadi, 1996, 15)

b) LENGUAJE DE PROGRAMACIÓN PYTHON

El lenguaje de programación Python es muy interesante, debido a su simpleza y claridad., es también un lenguaje poderoso que puede ser usado para administrar sistemas y desarrollar grandes proyectos. Es un lenguaje claro y objetivo, pues va directamente al desarrollo de la solución. (Coutinho Menezes, 2017).

c) DESCRIPCIÓN GENERAL DE LA HERRAMIENTA

El bot conversacional está desarrollado con Microsoft Bot Framework y con la API de Cognitive Services language understanding intelligent services (la cual implementa Machine Learning). Además el gestor de base de datos a utilizar es MySQL, como se muestra en la siguiente figura:

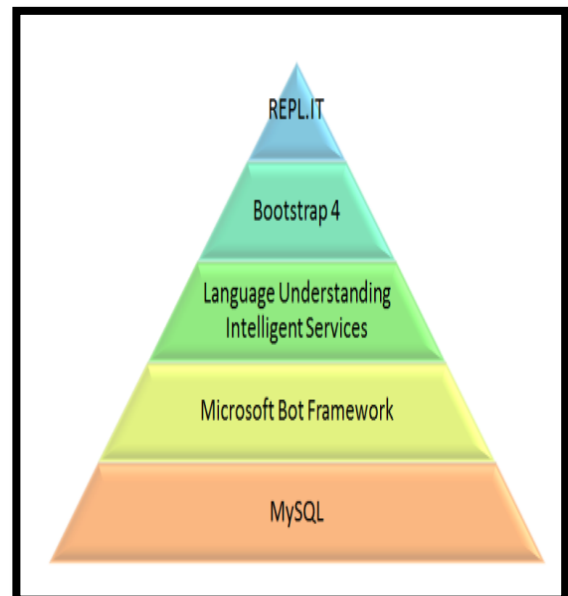


Figura 1.- Presenta la arquitectura de la solución con el cual se logra la integración e interacción de parte del estudiante con la parte inteligente de la herramienta.

La plataforma virtual “repl.it” es la que permite editar y ejecutar código Python mediante una consola online, en esta plataforma el usuario escribe su código y si tiene dudas sobre la sintaxis del lenguaje python entonces le pregunta al bot conversacional.

a) DIAGRAMAS:

A continuación, se muestra el diagrama de caso de uso del sistema para indicar cómo interactúan los diferentes componentes del sistema:

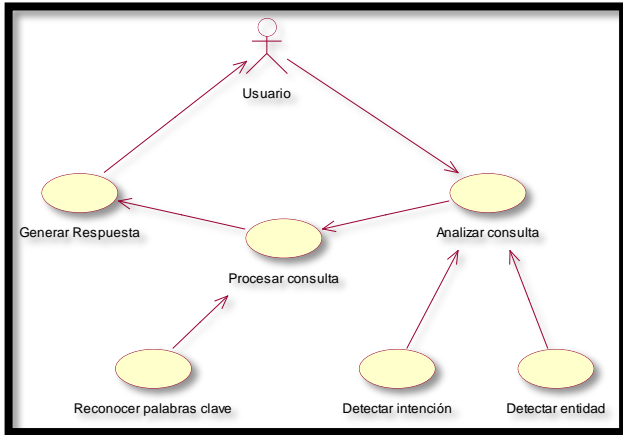


Figura 2.- Presenta el diagrama de casos de uso, donde se observan los diferentes actores del sistema y la forma como interactúan con énfasis en el usuario y el componente inteligente.

En la siguiente figura se muestra el diagrama entidad relación de los componentes del sistema.

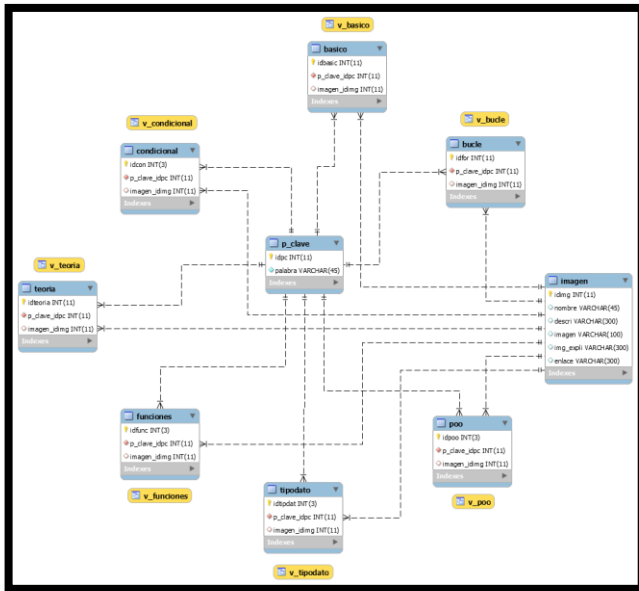


Figura 3.- Presenta el diagrama entidad relación, indicado la forma como interactúan los diferentes componentes del sistema.

Arquitectura de la solución al estilo de capas, permitiendo la intención con el sistema, estando en línea pudiendo acceder a nivel nacional, soportando diferentes tipos de conexiones incluyendo conexión rural, con lo cual permite tener una inclusión social a nivel educativo.

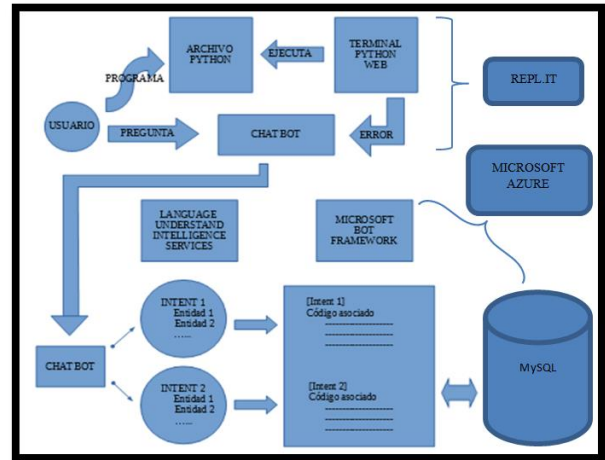


Figura 4.- Presenta la arquitectura del sistema indicando el flujo de la consulta y como fluye la consulta y la ejecución del código.

a) MACHINE LEARNING Y REGLAS DE PRODUCCIÓN

A continuación, se muestran algunas reglas usadas por la herramienta:

En la API language understanding intelligence services se ha creado un total de 9 intenciones, las cuales son Bucle, condicional, funciones, none, poo (se refiere a programación orientada a objetos), saludo, teoría (se refiere a preguntas sobre que es python o sobre porque se dice que python es un lenguaje interpretado), tipodato (se refiere a como declarar variables string, int, etc.) y “topicosbasicos”, cada intención tiene un conjunto de enunciados, en dichos enunciados pueden existir entidades (estas ayudan a reconocer la intención del usuario.)

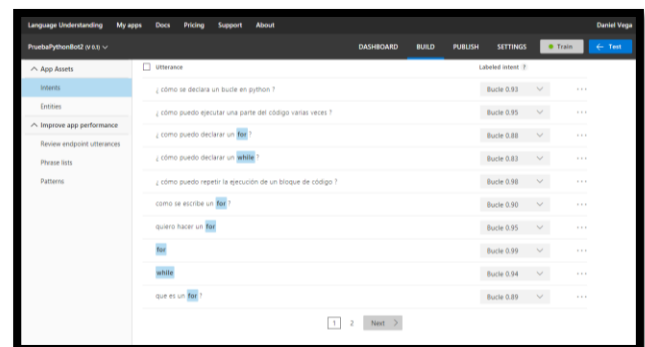


Figura 5.- Presenta el conjunto de enunciados (utterances) correspondiente a la intención (intent) “Bucle”.

Una vez detectada la intención del usuario se activa una de las reglas correspondientes a dicha intención, a dicha regla le corresponde una tabla de la base de datos (además a una misma tabla le pueden corresponder varias reglas).

```
[LuisIntent("Condicional")]
public async Task CondicionalIntent(IDialogContext context, LuisResult result)
{
    string consulta = "SELECT palabra,nombre,descri,imagen,enlace FROM v_condicional;";
    await ConsultarBD2(context, result, consulta);
}

[LuisIntent("Bucle")]
public async Task BucleIntent(IDialogContext context, LuisResult result)
{
    string consulta = "SELECT palabra1,nombre,descri,imagen,enlace FROM v_bucle;";
    await ConsultarBD2(context, result, consulta);
}

[LuisIntent("Funciones")]
public async Task FuncionesIntent(IDialogContext context, LuisResult result)
{
    string consulta = "SELECT palabra,nombre,descri,imagen,enlace FROM v_funciones;";
    await ConsultarBD2(context, result, consulta);
}
```

Figura 6.- Presenta las reglas y las tablas relacionadas a las intenciones creadas en la plataforma LUIS.

A continuación se hace la consulta y se obtienen todos los campos de la tabla, luego se hace uso de la entidad asociada a la intención del usuario para compararla con el campo palabra clave de la tabla correspondiente. En caso no hubiera una entidad se muestra un mensaje por defecto relacionado con la intención del usuario.

```
public async Task ConsultarBD2(IDialogContext context, LuisResult result, string consulta)
{
    using (SqlConnection conn = ObtenerConexion())
    {
        await conn.OpenAsync();
        using (var command = conn.CreateCommand())
        {
            string[,] p1 = new string[10, 5];
            command.CommandText = consulta;
            using (var reader = await command.ExecuteReaderAsync())
            {
                int i = 0;
                while (await reader.ReadAsync())
                {
                    p1[i, 0] = reader.GetString(0);
                    p1[i, 1] = reader.GetString(1);
                    p1[i, 2] = reader.GetString(2);
                    p1[i, 3] = reader.GetString(3);
                    p1[i, 4] = reader.GetString(4);
                    i++;
                }
            }
            await ShowLuisResult2(context, result, p1);
        }
    }
}
```

Figura 7.- Presenta el código de programación que permite la conexión con la base de datos así como procesar la consulta seleccionada de acuerdo a la intención del usuario.

El bot conversacional siempre responde con una imagen de manera que el usuario reciba una respuesta visual que le permita entender rápidamente los conceptos básicos de python.

```
private async Task ShowLuisResult2(IDialogContext context, LuisResult result, string[,] p)
{
    var reply = context.MakeMessage();
    EntityRecommendation entity = null;
    for (int i = 0; i < p.Length / 5; i++)
    {
        result.TryFindEntity(p[i, 0], out entity);
        if (entity != null)
        {
            if (entity.Type == p[i, 0])
            {
                reply.AttachmentLayout = AttachmentLayoutTypes.Carousel;
                reply.Attachments = GetCards(p[i, 1], p[i, 2], p[i, 3], p[i, 4]);
                await context.PostAsync(reply);
                context.Wait(MessageReceived);
                break;
            }
            else if (i == p.Length / 5 - 1)
            {
                reply.Attachments.Add(GetImageCard(p[0, 1], p[0, 2], p[0, 3], p[0, 4]));
                await context.PostAsync(reply);
                context.Wait(MessageReceived);
                break;
            }
        }
    }
}
```

Figura 8.- Presenta el código necesario para mostrar una imagen con ejemplos sencillos para que el usuario pueda aprender la sintaxis de python.

A continuación, se presenta las herramientas utilizadas:

Nombre	Tipo de tecnología	Versión	Utilidad
REPLIT	Plataforma online para programar en más de 40 lenguajes.	1.0	Necesario para ejecutar a través de una consola online los archivos que el usuario creará en python.
Bootstrap	Framework para diseño web responsive.	4.0	Permite crear páginas web con un diseño profesional en menos tiempo.
Language Understanding Intelligent Services	API de Cognitive Services		Permite la implementación de Machine learning en los chat-bots.
Microsoft Bot Framework	Framework para desarrollo de chat-bots		Framework utilizado para la implementación de un CHATBOT.
MySQL	Sistema gestor de base de datos relacional	5.7	Permite guardar las respuestas que serán utilizadas frente a las preguntas que el usuario realice.

Figura 9.- Presenta la lista de las herramientas utilizadas para el desarrollo de la solución.

III. RESULTADOS

Los resultados se presentan en dos aspectos, primero como resultado de la interacción de los alumnos con la herramienta y segundo presentando las mejoras de las habilidades de los alumnos con respecto al desarrollo de las habilidades lógicas de programación.

Presentamos una muestra de la interfaz del sistema, la cual consta de un textarea en donde el usuario podrá insertar su código Python, a la derecha del textarea está la consola en donde se muestran los resultados, esta parte pertenece a la plataforma “repl.it”, debajo está el bot conversacional en el cual el usuario podrá realizar las consultas necesarias.

Ejemplo 1: Programa que de 3 números identifica el mayor y el menor realizado con estructuras de control condicionales (IF-ELSE) y el uso de funciones:

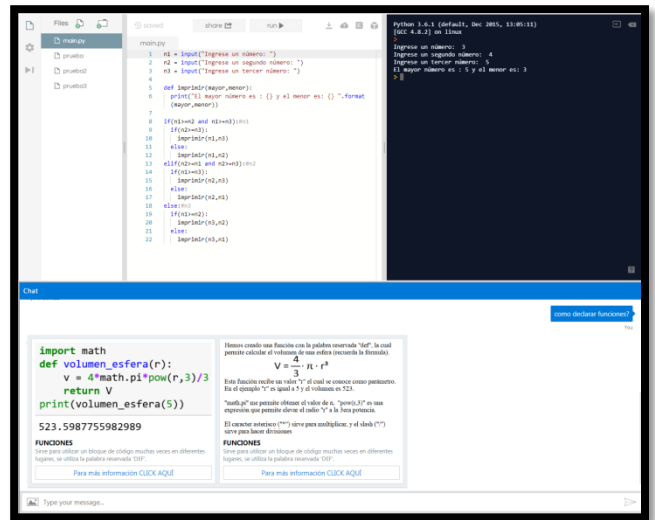


Figura 10.- Ejemplo matemático donde se requiere identificar el menor y mayor número usando sentencias if- else y funciones.

Ejemplo 2: Programa que calcula la suma de los “n”

primeros números naturales haciendo uso del bucle “FOR”:

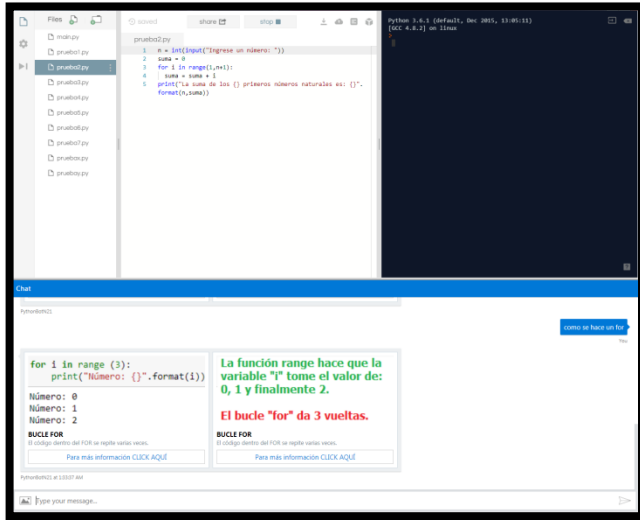


Figura 11.- Ejemplo matemático donde se calcula la suma de “n” primeros números naturales, mediante el uso de bucle “FOR”.

EJEMPLO 3: Programa que permite calcular el producto de todos los números que van desde el primero al segundo, haciendo uso de la estructura de control IF-ELSE y del bucle “WHILE”:

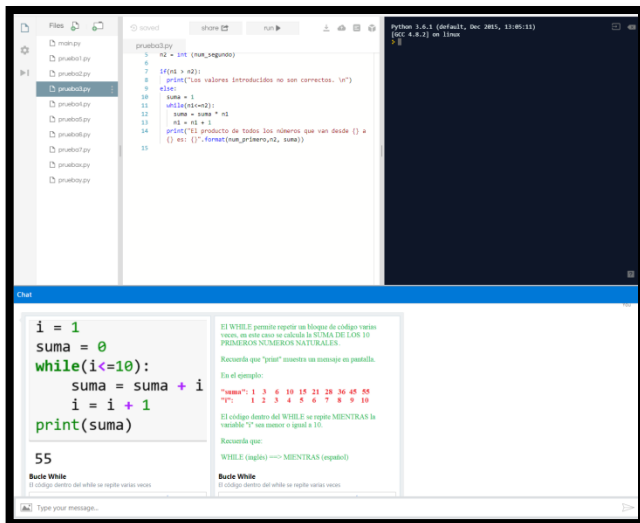


Figura 12.- Ejemplo matemático donde se calcula el producto de dos números mediante uso de la sentencia “WHILE”.

Uno de los problemas mayores al momento de aprender a programar es conocer la sintaxis de los bucles, en los siguientes ejemplos se presentan como usar las sentencias de control en Python.

EJEMPLO 5: Programa permite usar la sentencia “WHILE”

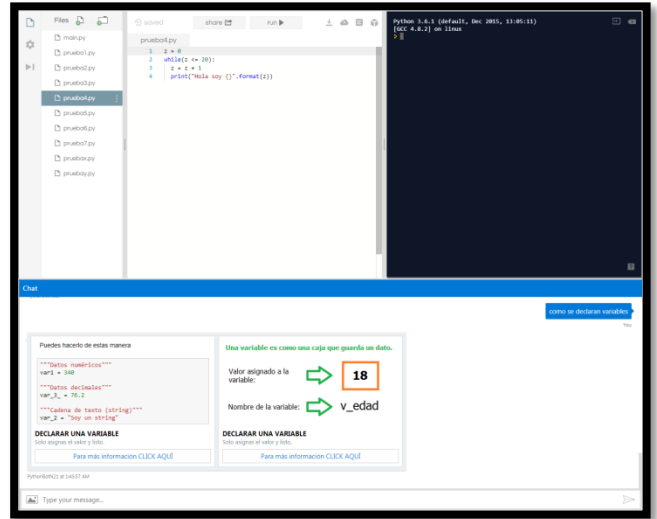


Figura 13.- Ejemplo en donde el bot explica cómo declarar una variable.

El manejo de las funciones es importante, la sentencia condicional “if”, es la más usada para poder preguntar alguna condición, presentamos un ejemplo de cómo usar la sentencia “if”.

EJEMPLO 6: Programa permite usar la sentencia “IF - ELSE”:

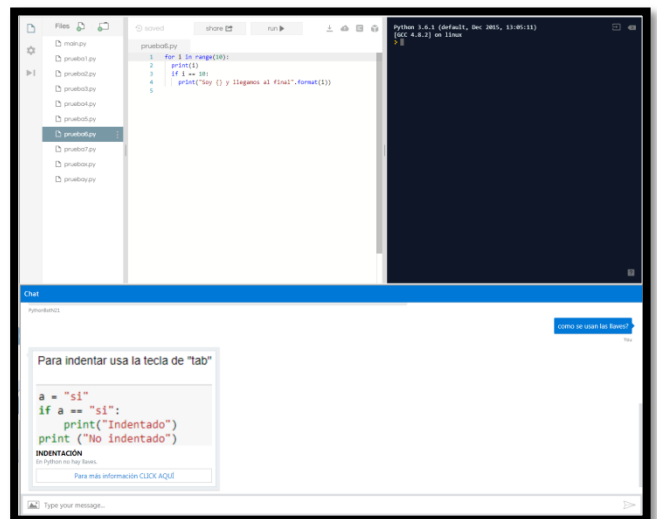


Figura 14.- Ejemplo de sintaxis donde explica cómo se aplica el concepto de sangría en una sentencia if.

Luego de realizar los procesos y las operaciones necesarias es necesario mostrar el resultado, para ello es común utilizar una función de impresión en pantalla, en Python utilizamos la función “print”, en la siguiente figura se presenta un ejemplo de uso de la función “print”:

EJEMPLO 7: Programa permite usar la función “PRINT”:

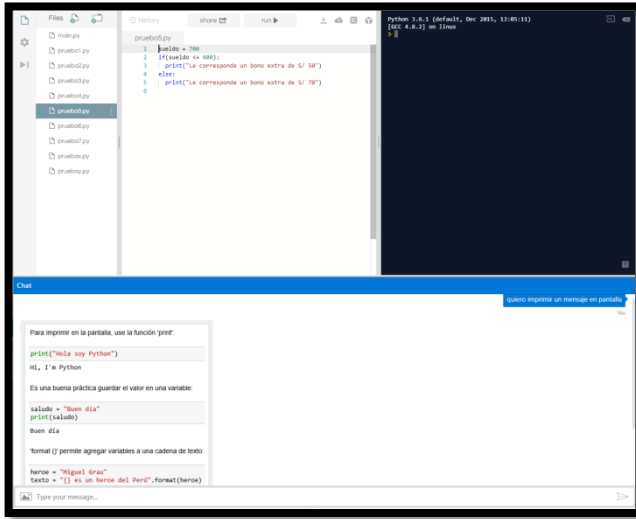


Figura 15.- Ejemplo de sintaxis donde explica cómo se usa la función “PRINT” usada para imprimir en pantalla.

Para los resultados donde se analiza las habilidades que desarrollaron los alumnos luego llevar un curso de programación básica e interactuar con la herramienta presentada, las pruebas consistieron en presentar a los alumnos una prueba de entrada antes de llevar el curso e interactuar con la herramienta, luego de finalizado el curso se toma otra prueba donde se evalúa la influencia del curso y de la herramienta, los indicadores que fueron evaluados fueron: resolución de ejercicios de razonamiento matemático, habilidades lógicas de programación y conocimiento de sintaxis de programación. A continuación, se presenta los resultados de las evaluaciones:

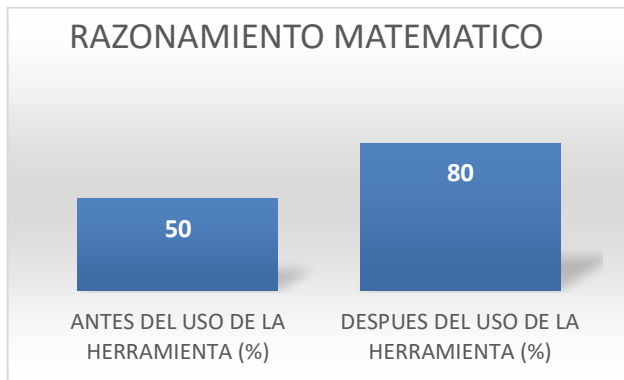


Figura 16.- Presenta el indicador razonamiento matemático, antes del uso de la herramienta los alumnos presentaban poco interés en poder solucionar problemas usando otras técnicas, por ello tenemos un 50% con el uso de la herramienta el interés y las habilidades presentaron un incremento al 80% logrando aumentar un 30% debido a la herramienta despertó en los alumnos el interés por solucionar problemas con mayor grado de complejidad.

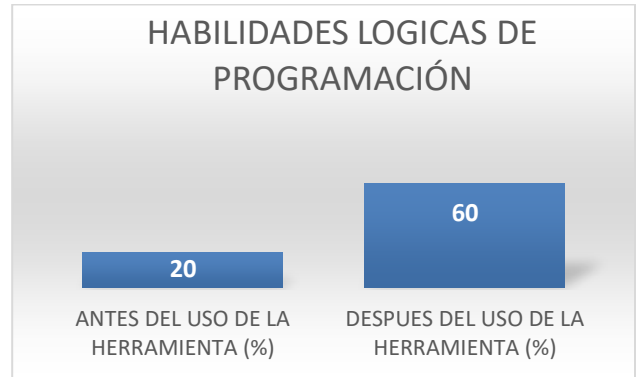


Figura 17.- Presenta el indicador habilidades lógicas de programación, antes del uso de la herramienta los alumnos presentaban dificultades para poder abordar la resolución de problemas por ello el resultado muestra un 20% en plantear un problema usando lógica de programación, solo conociendo como es el flujo de datos en una estructura de programación, con el uso de la herramienta las habilidades aumentaron a un 60% debido a que conocieron como es el flujo de datos cuando se programa y también gracias a los ejemplos desarrollados en las sesiones de clase, conforme se presenten más ejemplos con problemas más complejos esas habilidades aumentaran.

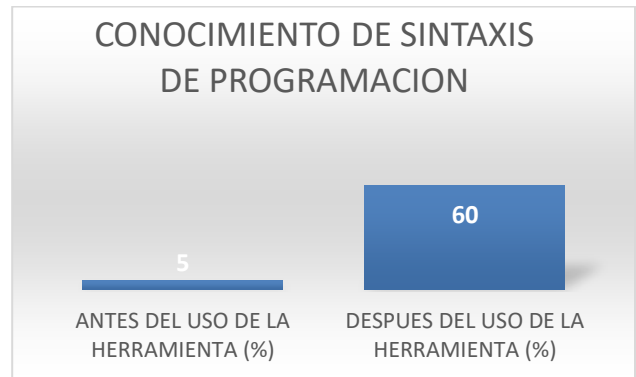


Figura 18.- Presenta el indicador conocimiento de sintaxis de programación, antes del uso de la herramienta los alumnos presentaban un desconocimiento de la sintaxis así como el uso de las palabras reservadas en programación por ello el resultado muestra un 5% antes de la prueba se realizó una explicación de cómo funcionan los lenguajes de programación por ello el 5%, con el uso de la herramienta los conocimientos de la sintaxis aumento considerablemente hacia un 60% debido a que conocieron como la sintaxis y el uso correcto de las palabras reservadas, conforme se usan librerías para tareas más complejas el conocimiento de la sintaxis crecerá a la misma proporción.

IV. CONCLUSIONES

La herramienta contribuye a que los alumnos puedan mejorar su nivel de percepción en el uso de los lenguajes de programación y con ello en mejorar las habilidades al momento de abordar problemas matemáticos, la herramienta permitió también conocer cómo se programa en Python debido que a través de las consultas la herramienta le indica cómo se usa cada función, palabra reservada y sentencia de control, también proporcionando el tipo de error y como poder solucionar, con ello se demuestra que “se aprende a programar solo programando”. Con ello se despierta el interés por parte de los alumnos en ingresar al mundo de la programación y poder explorar los recursos computacionales que hoy en día se tiene disponible.

El curso duro un promedio de 2 meses con una frecuencia de 3 horas académicas por semanas, en la institución donde se realizó el curso se recomendó poder incorporar el curso en sus sílabos del curso de computación para poder tener una mayor cantidad de horas para poder mejorar las habilidades de los alumnos, como también la herramienta está disponible para poder ser implementada en cualquier institución educativa a nivel nacional.

A nivel del uso de la herramienta el usuario pueda corregir con mayor facilidad errores de sintaxis y errores en tiempo de ejecución (por ejemplo, errores aritméticos, errores en la declaración de nombres de variables o atributos de clases). La herramienta le muestra el tipo de error y como puede corregirlo.

Una de las ventajas por el cual el uso de la herramienta se considera de utilidad, es la portabilidad y por ser multiplataforma, debido que la herramienta ejecuta en cualquier dispositivo como, por ejemplo: PC, Tablet, Celular, entre otros indistintamente del sistema operativo, solo se requiere una conexión a internet para poder acceder a la herramienta.

A nivel de la metodología se logro demostrar que trabajando con la motivación del estudiante logramos interactuar mejor y despertar más el interés en las matemáticas y las ciencias de la computación, porque logramos demostrar que cosas que parecen difíciles de resolver, se pueden dar solución usando la lógica de programación, la neuroeducación nos permite abordar el aprendizaje desde el punto de vista de como trabaja el cerebro, para poder utilizar estas características en el diseño de nuevas estrategias de aprendizaje.

REFERENCIAS

- [1] About Language Understanding (LUIS)
<https://docs.microsoft.com/en-us/azure/cognitive-services/LUIS/Home>
- [2] Ballestra M, Martínez J, Argibay P. “Matemáticas y cerebro”. Revista del Hospital Italiano de Buenos Aires 26, 2006.
- [3] Create a bot with the Bot Builder SDK for .NET
<https://docs.microsoft.com/en-us/bot-framework/dotnet/bot-builder-dotnet-quickstart>
- [4] Dehaene S, Spelke E, Pinel P, Stanescu R, Tsivikin S. “Sources of mathematical thinking: behavioral and brain-imaging evidence”. Science 284, 1999.
- [5] E. Castillo, J. M. Gutiérrez & Ali S. Hadi, Expert Systems and Probabilistic Network Models, New York: Springer-Verlag Pub., 1997.
- [6] Fernández J. “Neurociencia y enseñanza de la matemática”, Revista Iberoamericana de Educación 51, 2010.
- [7] Microsoft Bot Framework .NET - Your first Bot
<https://www.youtube.com/watch?v=zLcFx6qcMQE&list=PLgF-CyaX1p3FE55OTRNH-kOb16zqeBZCo>
- [8] N. N. Countinho, Introducción a la programación con Python, Sao Paulo: Novatec Pub., 2017.
- [9] Radford L, André M, “Cerebro, cognición y matemáticas”, Revista Latinoamericana de Investigación en Matemática Educativa 12, 2009.
- [10] Sajjad Hosseini - Chat Bots Implementation and User Engagement
https://www.theseus.fi/bitstream/handle/10024/126000/Hosseini_Sajjad.pdf?sequence=1&isAllowed=y
- [11] TI Capacitación (2018) <https://ticapacitacion.com/cursos/botses>
- [12] welcome to puthon.org. <https://www.python.org/>