



*Asesorías y Tutorías para la Investigación Científica en la Educación Puig-Salabarría S.C.
José María Pino Suárez 400-2 esq a Lerdo de Tejada, Toluca, Estado de México. 7223398475*

RFC: ATI120618V12

Revista Dilemas Contemporáneos: Educación, Política y Valores.

<http://www.dilemascontemporaneoseducacionpoliticayvalores.com/>

Año: VII Número: Edición Especial Artículo no.:8 Período: Noviembre, 2019.

TÍTULO: Estudio y propuesta metodológica para la enseñanza-aprendizaje de la programación informática en la educación superior.

AUTORES:

1. Máster. Rosa Elizabeth Molina Izurieta.
2. Máster. Renzo Rogelio Padilla Gómez.
3. PhD. Maikel Yelandí Leyva Vázquez.

RESUMEN: El estudio de la programación informática requiere el desarrollo o la explotación de habilidades lógicas y de razonamiento para la investigación y/o desarrollo de nuevos productos dentro de la informática. Experimenta desafíos de cómo los estudiantes llegan a comprender los procesos de creación y diseño de un software, creando patrones empíricamente efectivos de desarrollo de programas y que resultan en materia prima para poder crear nuevas metodologías. El aprendizaje de un lenguaje de programación puede no ser una tarea sencilla porque implica el razonamiento del paradigma de programación en el que se enmarca, la comprensión de la sintaxis del lenguaje en sí, el desarrollo de una lógica de programación correcta y la aplicación de una buena metodología de desarrollo de software, todo al mismo tiempo.

PALABRAS CLAVES: Enseñanzas-aprendizaje, programación informática, Educación Superior, lenguaje de programación.

TITLE: Study and methodological proposal for the teaching-learning of computer programming in higher education.

AUTHORS:

1. Máster. Rosa Elizabeth Molina Izurieta.
2. Máster. Renzo Rogelio Padilla Gómez.
3. PhD. Maikel Yelandi Leyva Vázquez.

ABSTRACT: The study of computer programming requires the development or exploitation of logical and reasoning skills for the research and/or development of new products within computer science. It experiences challenges of how students come to understand the processes of creating and designing a software, creating empirically effective patterns of development of programs and that result raw material to be able to create new methodologies. The learning of a programming language may not be a simple task because it implies the reasoning of the programming paradigm in which it is framed, the understanding of the syntax of the language itself, the development of a correct programming logic and the application of a good software development methodology, all at the same time.

KEY WORDS: Teaching-learning, computer programming, Higher Education, programming language.

INTRODUCCIÓN.

La comprensión de la programación informática implica diseñar, estructurar y utilizar metodologías soportadas por herramientas TIC usado no solo por el docente, sino también para el ciudadano con conocimientos informáticos (Crow, T., Luxton, A. y Wuensche, B. 2018).

El docente tiene la responsabilidad de conocer la o las metodologías y herramientas TIC que permiten transmitir y desarrollar en los estudiantes sus capacidades de comprensión lógica y razonamiento para la resolución de algoritmos que es donde llevará este estudio científico.

Este proyecto pretende crear o fortalecer metodologías para la enseñanza plena de la programación informática de manera que, se potencie las habilidades que el estudiante necesita para comprender y solucionar algoritmos teniendo como recurso el desarrollo de la lógica y razonamiento y más aún amparados por una herramienta idónea de software para estos propósitos. Así mismo, docentes podrán disponer de material de primer nivel para poder impartir con mayor facilidad sus clases de programación informática.

¿Como lograr todo esto? Se cree que puede existir un desorden en la secuencia de enseñanza de la programación lo que conlleva a pensar en una nueva metodología basada en “aprendizaje sistematizado por niveles” la cual encontrará y cubrirá las deficiencias en el proceso según se avance. A esta propuesta puede sumarse como complemento una estructura metodológica que domine el enfoque problémico que será idónea para identificar y fundamentar los motivos del porqué de las deficiencias en el aprendizaje de la programación informática de los alumnos de nivel superior.

Otras metodologías como soporte que se consideran son, el aprendizaje activo que orienta y alienta a los estudiantes a tener una actitud comprometida y de responsabilidad en su proceso de aprendizaje (Rosique, F. C., Losilla F. L y Pastor J. F., 2019). Esto es muy importante porque la lógica y el razonamiento para la solución de problemas algorítmicos, se van ganando progresivamente.

Los resultados están encaminados como primera instancia a encontrar y fundamentar las falencias que tienen los estudiantes para la programación informática, obtener estudiantes con habilidades y destrezas para la programación usando las técnicas necesarias junto con las herramientas de software adecuadas que permitan resolver todo algoritmo presentado, es el resultado más ambicioso a

perseguir. Por último, Docentes plenamente capacitados con los conocimientos necesarios en materia de metodologías y técnicas eficientes para impartir sus cátedras de programación.

DESARROLLO.

Se presentan los objetivos.

Objetivo General: Proponer una metodología y herramienta con técnicas innovadoras para la enseñanza de la programación informática en la Educación Superior, que proporcione tanto a estudiantes como docentes las habilidades y destrezas para desarrollar y solucionar algoritmos informáticos.

El proyecto aborda el problema y dificultades que tienen los estudiantes en primera instancia para comprender la lógica y razonamiento que exige la programación informática así mismo el vacío que puede tener un Docente para impartir correctamente sus clases de programación de manera que, al término de cada sesión de clases quede comprendido. A continuación, se exponen algunos de los trabajos más relacionados con la investigación propuesta en este proyecto indicando cuáles son sus deficiencias con la propuesta del proyecto.

Según el trabajo de Timarán, R., Toledo, J. y Chávez, A. (2012), Apropiar un lenguaje de programación multiparadigma en la formación de los ingenieros de sistemas llevaría a fortalecer el estudio de las diferentes metodologías de programación en lugar de dedicar basto tiempo determinado en la enseñanza del lenguaje formal para luego aplicar a un paradigma determinado. Además, los estudiantes de Ingeniería de Sistemas desarrollarían la competencia de integrar los diferentes paradigmas con el fin de construir mejores productos de software.

Ahora de acuerdo al estudio realizado por López, J., Hernández, C. y Farran, Y. (2011), en los cursos introductorios de programación, los profesores inician a los estudiantes en la disciplina de computación y en el proceso de resolución de problemas algorítmicos. Aprender a programar es una

tarea compleja que requiere, primero, entender el problema y luego diseñar un algoritmo que represente los pasos de solución para implementarlo en un lenguaje de programación.

La construcción de la solución en un lenguaje de programación es un proceso cíclico que consiste en codificar, compilar, probar y depurar programas. Para comprobar que un programa está correcto, éste debe pasar exitosamente varios casos de prueba. Cada caso de prueba consiste en ejecutar el programa usando un conjunto de datos de entrada y luego analizar los resultados.

De acuerdo a Vidal, C., Cabezas, C., Parra, J. y López, L. (2015), es relevante para el ser humano, el desarrollo del pensamiento lógico para analizar y solucionar problemas y situaciones de su vida diaria. Particularmente, su uso es esencial para el análisis, comprensión y solución de problemas computacionales, relacionados con la cultura de nuestra época y, por lo mismo, es imperativo su desarrollo en las presentes y futuras generaciones. En este aspecto, el desarrollo del pensamiento lógico y algorítmico es, usualmente, mediado por el uso de lenguajes de programación cuyo aprendizaje requiere un dominio de la sintaxis y semántica de dichos lenguajes, hecho que dificulta el acceso y el uso de los sistemas computacionales.

Por otra parte, Muñoz, R., et al. (2015), menciona la introducción a los lenguajes de programación es un curso de suma importancia para cualquier carrera asociada a las Ciencias de la Computación. Sin embargo, en el primer año de estos programas, estudiantes y sus profesores se ven envueltos en graves problemas. La aplicación de conceptos básicos o el diseño de algoritmos que son relativamente simples para los docentes, parece ser algo difícil para el estudiante. La mayor parte de esos problemas son originados por la complejidad de los conceptos tales como variables, estructuras repetitivas, arreglos, funciones de los lenguajes de programación.

De acuerdo a estudios de Zuleta, A. y Chávez, A. (2011), la dificultad en el aprendizaje de la programación se evidencia en el bajo promedio que obtienen los estudiantes en asignaturas de programación y en el alto índice de los que las repiten, quienes, superan el 26%. Aunque han sido

desarrollados abundantes estudios sobre el tema, al parecer aún no se logra una solución satisfactoria, quizá porque los trabajos se ocupan de ciertos elementos curriculares, pero no de la integralidad del problema.

Las investigaciones y textos revisados y más aún la experiencia en el área de informática, da autoridad para decir que, la enseñanza de la programación no se limita solamente a la explicación de pasos, esquemas y de conceptos teóricos todo esto, va más al fondo persigue despertar o en su defecto potenciar habilidades y destrezas en base a la lógica que intervienen naturalmente en el desarrollo y solución de algoritmos usando conceptos abstractos, como por ejemplo la manipulación de datos, sintaxis, semántica y hasta el metalenguaje. Se sabe que la programación es compleja, existen debilidades en los educandos y en los educadores la experiencia a veces no es válida si se tiene la metodología correcta punto importante de este estudio.

Definición del problema y justificación de la investigación.

Problema.

Como se conoce del tema, se sabe que la programación es una actividad compleja que requiere inteligencia, conocimiento, habilidad y disciplina, las cuales sólo son desarrolladas con el paso del tiempo y la práctica constante. Quizás el Docente es un perito en la materia, pero, su capacidad de transmitir el conocimiento al estudiante puede no ser el idóneo por desconocimiento de una metodología y técnicas que permitan al alumno asimilar y comprender con rapidez la lógica de programación.

La falta de ese desarrollo de la lógica y razonamiento para la programación que el estudiante debe poseer, obliga a identificar esos obstáculos dentro del proceso de enseñanza-aprendizaje, tratando de encontrar las actividades que implican mayor dificultad entre ellas el porqué de cada instrucción, la prueba de escritorio, la identificación de errores y la corrección de los mismos entre otros.

Para poder programar es necesario desarrollar habilidades lógicas y de razonamiento, se debe conocimiento del lenguaje, del entorno de programación y dominio de los conceptos y técnicas propias de la programación. Para los que no están inmersos de lleno en el problema, suelen apuntar solo a las debilidades del estudiante. Y es que el problema no sólo pasa por el estudiante, pues el docente juega un papel importante en el proceso de aprendizaje de la programación informática, si él trabaja de forma empírica sin un sustento esquemático o metodológico y más aún sin respaldo de las tecnologías, el resultado puede ser tan desastroso que puede obligar a la deserción del estudiante que no verá progreso en su aprendizaje.

Esto lo puntualiza Arellano, J., Nieve, O., Arista, G. y Solar A. (2011) quién dice que, una de las formas habituales de enseñar a programar es forzar al estudiante a resolver un gran número de ejercicios y problemas con el método de codificar, probar y corregir hasta lograr que el programa produzca los resultados correctos. En este esquema de enseñanza-aprendizaje se deja de lado la importancia de entender a cabalidad un problema para concebir un algoritmo de solución, esto a su vez ocasiona que más que formar programadores se estén formando codificadores.

Desde el inicio del proceso de aprendizaje de la programación, Spigariol, L. (2015), menciona que es claro en decir que el problema al que el estudiante se enfrenta, es la rigurosidad del lenguaje de programación teniendo como herramienta su lenguaje natural cargado de sobreentendidos y ambigüedades, su acalorado entusiasmo o desazón, cuando las cosas no salen se choca con la frialdad inapelable de la computadora, partiendo de la multiplicidad de formas en que las personas se interpreta la información se encuentra con lo acotado de los tipos de datos que disponen los lenguajes de programación.

En todos los centros de estudios superiores que cuentan con la enseñanza de programación informática, inician sus labores académicas tratando de que, al culminar el pensum de estudios, los estudiantes obtengan un porcentaje elevado en el dominio de la programación. A pesar de todos los

esfuerzos del cuerpo docente, no se logra un resultado acorde al esperado, los factores son diversos y están en el orden de experiencia docente, falta de metodologías apropiadas, técnicas efectivas. Creemos que existe un desorden en la secuencia de aprendizaje apoyados en el uso de herramientas de software.

Una propuesta para ayudar a resolver esta problemática es implementar según Del Prado, A. y Lamas, N. (2014), el uso de software orientado a la resolución de problemas; que enseñen a los alumnos a seguir un proceso (pensamiento algorítmico) y que contenga actividades de autoevaluación que se imparten a través de la plataforma moodle del aula virtual. Esto a nuestro juicio se parece a un esquema ordenado de aprendizaje.

Se persigue las causas y problemáticas del porqué de las dificultades para programar por ello, Insuasti, J. (2016), manifiesta que, existen estudiantes que no logran adquirir las habilidades necesarias para la programación, incluso después de la terminación de un curso de fundamentos de programación en las Ciencias Computacionales. Además, menciona que algunas investigaciones indican que las razones para no lograr los niveles de aprendizaje deseados, pueden ser debido a la complejidad de la sintaxis del lenguaje y los conceptos de programación; la carga cognitiva implicada en el aprendizaje de programación; el mal diseño de los objetos de saber, y la falta de habilidades cognitivas propias para la solución de problemas.

Justificación.

En bien conocido las dificultades que tiene las persona en común o estudiante, el inicio de aprendizaje de la programación informática en la lógica y el razonamiento. De manera parecida, para el docente representa mayores esfuerzos que concentrará recursos para satisfacer el conocimiento de esos estudiantes con dichas deficiencias.

Este proyecto es justificable porque permitirá revelar nuevas metodologías que sustentarán la lógica y razonamiento en la resolución de problemas de programación lo que contribuirá a la formación de una cultura algorítmica y a perfeccionar las habilidades para programar de los estudiantes de la Ciencia de la Computación e Ingeniería Informática.

Un estudio de esta magnitud, es justificable porque permitirá encontrar las causas y efectos que ocasionan las dificultades que tienen los estudiantes para aprender a programar y solucionar problemas algorítmicos.

No contar con los recursos necesarios sean teóricos, tecnológicos sumado a docentes que no están preparados mucho menos poseen una línea metodológica acorde, generan estudiantes con muy bajo nivel de destrezas en solucionar problemas algorítmicos.

Así lo expone Zuleta, A. y Chávez, A. (2011) donde menciona, que la dificultad en el aprendizaje de la programación se evidencia en el bajo promedio que obtienen los estudiantes en asignaturas de programación y en el alto índice de los que las repiten, quienes, en algunos estudios superaron el 26%. Y aunque han sido desarrollados abundantes estudios sobre el tema, al parecer aún no se logra una solución satisfactoria, quizá porque los trabajos se ocupan de ciertos elementos curriculares, pero no de la integralidad del problema; y es que la complejidad que los estudiantes se enfrentan en el aprendizaje de la programación informática, va más allá de herramientas y conceptos que no pueden justificar del todo el aprendizaje. Así lo menciona Insuasti, J. (2016), que hace referencia al siguiente análisis “La mayoría de los estudiantes, encuentran difícil y compleja la tarea cognoscitiva relacionada a la programación de computadoras” y explicaron: “el aprendizaje demanda complejas habilidades cognitivas tales como la planificación, razonamiento y resolución de problemas en programación de computadoras”.

Esto puede ser relevante para justificar un estudio de esta naturaleza pues, con esto se entiende que existen habilidades de pensamiento que, de no haber sido desarrolladas previamente, se constituyen en factores negativos para el aprendizaje del alumnado.

Metodología de la investigación.

Este trabajo tiene aristas bien definidas de investigación que se centran en el nivel progresivo y evolutivo de aprendizaje de la programación informática. Es al estudiante a quien se dirige mayormente la investigación y la metodología junto con las técnicas que se pretenden usar pues estas, irán asociadas a él para desarrollar o potenciar la lógica y razonamiento en la solución de problemas algorítmicos.

¿Pero cómo lograr que la lógica y el razonamiento fluya en el estudiante? Las diferentes metodologías y técnicas a usarse darán soporte a este objetivo que, junto con herramientas de software bien seleccionadas o diseñadas, se tendrá resultados favorables. El docente no deja de ser una pieza clave de este proceso y las metodologías serán aplicadas por ellos.

Son múltiples las metodologías que pueden aportar a este trabajo de investigación, pero, se persigue una estructura metodológica que domine el enfoque problémico en otras palabras, que el proceso de enseñanza-aprendizaje genere en los alumnos, esas capacidades de abstracción y razonamiento al enfrentarse a un problema algorítmico y a la vez, sea capaz de descubrir junto con los docentes, sus falencias y debilidades fortaleciéndolas.

Lo mencionado anteriormente es reforzado por Ocegüera, S., Expósito, C., Díaz, G. y Bonne, E. (2011), y menciona que debe predominar enfoque problémico en la metodología; es decir, un proceso de enseñanza-aprendizaje que se caracterice, por crear en los alumnos, de forma sistemática, la necesidad de tener que resolver problemas, tanto en la fase de obtención del contenido, como en la fase de su fijación.

La formación de profesionales informáticos de programación está exigiendo nuevas formas de enseñanzas. Las formas tradicionales o la aplicación de metodologías clásicas de enseñanza, no abastecen la formación de profesionales en la programación informática eficientes. A esta problemática, han nacido nuevas metodologías que pueden aportar a este objetivo tal como el aprendizaje activo y el aprendizaje basado en proyectos. A ello, Estévez, I., et al. (2013) señalan, que la metodología de aprendizaje activo y basado en proyectos “Permite adaptar de forma dinámica aspectos principales de una asignatura como temas, actividades de aprendizaje, configuración de grupos y actividades de evaluación”. Estas ideas pueden ofrecer pautas que permitan a los alumnos fortalecer el aprendizaje y trabajo individual o su vez puede representar una carga pesada debido a las múltiples de esta metodología.

Mucho se ha discutido sobre las diferentes formas y fuentes de aprendizaje, para un docente el encontrar material académico puede representar una tarea sencilla dada la riqueza de fuentes bibliográficas que existen ahora; sin embargo, la calidad de dicho material receptado puede marcar la diferencia al momento de impartir sus clases.

La programación informática es muy sensible en lo que respecta a material académico y varios autores han mirado a los artículos científicos como material de excelencia para la enseñanza y como metodología ya está mostrando el interés necesario. Así lo expone Trejos, O. (2013), quien “Proporciona una gran relevancia a los contenidos y a la manera como estos se presentan y, basado en ello, podría decirse que los artículos científicos constituyen buenas píldoras claves para promover y acceder al conocimiento de avanzada dentro de un formato coherente y accequible”.

Debido a la relevancia de la información que un artículo científico contiene y representa, esta metodología de su uso en la enseñanza de la programación, puede generar claves en el desarrollo en la formación de los estudiantes universitarios quienes pueden además desarrollar habilidades de investigador.

Si bien, las metodologías representan en centro de este estudio, las técnicas también abordan y enfrentan el problema de la enseñanza de la programación informática y son otra parte esencial dentro del proceso que estudiantes y docentes deben seguir. Múltiples son las técnicas aplicadas que según sus autores han dado frutos reflejados en alumnos con mejores capacidades para la programación.

Según Caneo, O., (2016), “La técnica AC se fundamenta en el aprendizaje y la interacción de estudiantes en pequeños grupos heterogéneos. Esta técnica ha demostrado su utilidad en el aprendizaje sobre resolución de problemas, aprendizaje de conceptos y en el desarrollo de la creatividad”.

Al parecer esta técnica se muestra muy interesante tal como lo mencionan Lovos, E., Gonzalez, A. y Bertone, R. (2013), quienes hacen referencia a lo siguiente: “En el ámbito educativo, las actividades de aprendizaje colaborativas buscan desarrollar en los alumnos un conjunto de habilidades que se relacionan en forma directa con el objetivo que persigue la educación moderna, la formación en competencias que le permiten al alumno integrarse en una esta nueva sociedad mediada por tecnologías digitales, donde el docente desde su lugar debe ser, dinamizador, orientador y asesor de todo el proceso de enseñanza y aprendizaje”.

Estos autores avalan que los ambientes colaborativos pueden ofrecer un importante soporte a los alumnos durante las actividades aprendizaje de la programación, además, agregan que la resolución de problemas a través de la colaboración promueve la reflexión, un mecanismo que estimula el proceso de aprendizaje.

Dada la naturaleza los objetivos planteados en la investigación, se puede considerar asumir una posición epistemológica naturalista, concretada en el paradigma socio-crítico la misma que estará respaldada bajo los preceptos del paradigma interpretativo–simbólico, por lo que se pretende comprender la hermenéutica que rodea a los actores del este estudio (estudiantes, docentes), inseparable de su trasfondo ideológico. Esta plataforma epistemológica proporciona herramientas

heurísticas que permitirán identificar las causas que ocasionan el bajo rendimiento en el aprendizaje de la programación informática de los alumnos de nivel superior.

Se pretende darle voz a aquellas situaciones de la vida diaria de quienes ejercen el papel de docente de informática que pueden pasar inadvertidas para las autoridades universitarias; sin embargo, son estas las que realmente dignifican la función, pero no son atendidos y existe escasa atención a la problemática.

Dado que el propósito de la investigación es comprender a fondo las debilidades en el proceso de enseñanza de la programación informática, la realidad que viven los estudiantes y docentes que se encuentran en contextos complejos pero relacionados, el paradigma socio-crítico permitirá comprender con visión global, de manera reflexiva y crítica, la práctica social de los estudiantes y docentes juntos a los condicionante a la hora de aplicar las metodologías propuestas. Estas debilidades pueden obedecer a múltiples circunstancias que van desde motivaciones hasta las mismas tareas que deben partir desde el aula de clase.

En este caso, Miños, A. (2016), menciona que “la ausencia de trabajos que aborden el trabajo de los conceptos básicos de programación desde estrategias de aula a nivel micro, es decir, la forma de presentar y trabajar dichos conceptos desde las distintas actividades de aula”. La intención es que, el trabajo de aula puede ser un apoyo importante a cualquier metodología aplicada.

Popkewitz, T. S. (2010) indica que la función socio-crítica es comprender las relaciones entre valor, interés y acción, que las condiciones estructurales y las acciones de los individuos influyen en una realidad social, modificándola. La cultura educacional es cambiante al ser construcción social en la que intervienen individuos que interactúan, se organizan y forman sus propios códigos para asegurar la convivencia pacífica de los involucrados en este tejido social mediatizados por fuerzas extra-organizativas de carácter ideológico.

Cada escuela posee una forma de organización y gestión escolar porque tiene una vida cotidiana propia y singular que le permite crear y recrear una cultura con identidad y autonomía, aunque esta última no es plena porque siempre esta mediatizada por las formas ideológicas que penetran en la escuela, provenientes del tejido social del que forma parte y muy particularmente de ese ámbito de decisiones que se denomina la macro política, Ball, J. (1989)

Investigar desde un paradigma socio crítico permite, como investigadora, aportar soluciones concretas para esa realidad con el protagonismo de los involucrados, por tener este paradigma sentido democrático del conocimiento, permitiendo que la investigación sea construida en y desde la realidad situacional, social, educativa y práctica de sujetos implicados en luchas, intereses, preocupaciones y problemas que forman parte de su experiencia cotidiana.

De acuerdo con la posición epistemológica asumida y justificada, la opción metodológica que se considera más adecuada para llevar a cabo la presente investigación es el método etnográfico. La etnografía es un recurso para explicar aquellas cosas que no tienen explicación racional, ayudando a desvelar cuestiones que para los involucrados en esos contextos pasaron desapercibidas; la describen como “espejo que permite ver mejor lo que realmente acontece en un contexto determinado” (Goetz, J. P. y LeCompte, M. D., 1988).

Con respecto a la etnografía, Sierra, F. S. (2006) subraya que como método de investigación opta por “conciliar saberes con comprensión, de manera que se alcance la emancipación de las personas a través del conocimiento científico de la cultura y la sociedad”.

La etnografía es importante por su valor holístico dentro de la metodología cualitativa, porque permite estudiar los fenómenos de manera integral, como un todo, en sus respectivos contextos. Es decir, la etnografía permite ver a los individuos que integran un ecosistema en el lugar donde ellos llevan a cabo las diferentes interacciones, actividades, construcción de valores, transmisión de ideologías,

generación de expectativas. El estudiante como integrante de ese escenario social es inseparable del mismo porque se estaría generando una.

La etnografía es una forma de estudiar la vida humana con estrategias que son empíricas y naturalistas; permiten la reconstrucción cultural, porque proporcionan datos fenomenológicos, empleando variedad de técnicas. Este método promete arrojar resultados novedosos, imposibles de obtener de ninguna otra manera. Por otra parte, debido a su flexibilidad, una investigación etnográfica permite la modificación de las estrategias y los objetivos en el proceso al presentarse nuevas dimensiones de estudio, limitaciones en el trabajo de campo o imprevistos. Posibilita partir de lo particular a lo general, favoreciendo el desarrollo una investigación exhaustiva con técnicas de recolección de datos específicas para lograr develar las categorías de análisis que emerjan de esos datos (Taylor, S. J., Bogdan, R. y Piatigorsky, J., 1987).

El estudio etnográfico que se pretende realizar se ajusta al diseño de estudio de caso único, dado que la investigación abarcará sólo a la Facultad de ciencias Matemáticas y Físicas de la Universidad de Guayaquil que tiene como denominador común su ubicación en contextos sociales diversos. Ahora bien, no se pretende un estudio comparativo de estas tres realidades, escolares y socio-educativas, porque desde la posición epistemológica adoptada se asume que son diferentes entre sí; aunque pueden tener concomitancias debido a la macro política que regula estas comunidades, tanto, en la faceta estrictamente educativa como en las dimensiones social, económica, cultural y urbanística.

Los tres comunidades en cuestión comparten estos marcos normativos, que gravitan sobre todos ellos de modo similar, y forman parte, en definitiva, de una misma sociedad que les impone una determinada manera de concebir el mundo institucional, unos referentes históricos concretos, unos universos simbólicos interiormente socializados Berger, P. L. y Luckmann, T. (1968) que desembocan en coincidencias, similitudes entre esas tres construcciones sociales que no son, pues, libremente hechas por sus respectivos actores.

Así pues, este estudio de casos múltiples permitirá detectar coincidencias y divergencias en el desempeño de la función directiva en contextos complejos; es decir, podrán identificarse las interpretaciones dadas por los directores a las políticas y programas educativos que son comunes y generales para el sistema educativo pero que, en cada contexto, tendrán su particular hermenéutico y, en consecuencia, su plasmación, diferencial o no, en la práctica.

La labor más compleja de la investigadora será la identificación, el afloramiento de los porqués que permitan comprender tanto los significados como las prácticas de los directores escolares en contextos caracterizados por su complejidad. El estudio de casos permite “interpretar los programas y políticas a través de las perspectivas de quienes lo aplican”, y los directivos son los que tienen la responsabilidad de poner en práctica las políticas y programas educativos, y aunque se dan unas directrices generales y homogeneizadoras para su aplicación, es bueno conocer cómo hacen la interpretación de dichas políticas y programas y cómo los adaptan a la realidad en la que ejercen su papel directivo.

En una primera fase de la investigación, se podrá aplicar la técnica del cuestionario de carácter cuantitativo para una aproximación a la realidad, que posteriormente se profundizará con la aplicación de técnicas antes señaladas, de manera persistente, hasta identificar y justificar las categorías de significado que permitan dar respuesta a los objetivos de la investigación.

Así mismo, se evaluarán otras metodologías que puedan aportar grandemente al proyecto así, [9] propone una metodología basada en el aprendizaje activo y el aprendizaje basado en proyectos para cursos de programación en un entorno universitario. Esta metodología permite adaptar de forma dinámica aspectos principales de una asignatura como temas, actividades de aprendizaje, configuración de grupos y actividades de evaluación.

El aprendizaje activo y el aprendizaje basado en proyectos son muy usados según Estévez, I., et al. (2013), se centran en el trabajo continuo en el día a día del alumno. Por un lado, ofrece pautas para que los alumnos organicen su tiempo, promoviendo el autoaprendizaje y el trabajo individual. Por otro, durante el proyecto los alumnos desarrollan sus capacidades de trabajo en equipo, fomentando el desarrollo de competencias transversales como el aprendizaje colaborativo.

Resultados esperados.

Usando las metodologías, técnicas y herramientas necesarias se ha logrado resultados alentadores en este campo.

La percepción que tienen los alumnos acerca del uso del software para desarrollar habilidades de programación informática, hace que García, N., Santiago, O. y Arellano, J. (2011) suponga, con cierta certeza, que la mayoría de estudiantes tienen un estilo de aprendizaje pragmático o activista; sin embargo, la percepción sobre su nivel de aprendizaje sugiere que el simple uso del software no es suficiente. Ha obtenido resultados donde el uso equilibrado de todos los elementos del esquema de aprendizaje para cubrir a todos los estilos, permite que los alumnos construyan su propio conocimiento y desarrollen las habilidades deseadas.

En cuanto a herramientas de software como soporte para potenciar la enseñanza de programación informática, Carreño, M., et al. (2015), menciona en su artículo “desarrollo de una herramienta web interactiva como apoyo para el diseño de algoritmos en pseudocódigo durante el proceso de enseñanza – aprendizaje”, tuvo como resultados: Los alumnos pudieron comprobar el funcionamiento paso a paso y el desempeño de los algoritmos de manera visual. La aceptación de la herramienta fue inmediata, hubo un gran interés por utilizarla para el desarrollo de otras prácticas. Se pudo notar que los alumnos mejoraban la habilidad para el desarrollo de algoritmos al presentar la ejecución de la misma de manera visual.

Estudios con plataformas de evaluación automática tal es el caso de López, J., Hernández, C. y Farran, Y. (2011), que incorpora mecanismos de reconocimiento de patrones de comportamiento y una metodología que incluye un conjunto de criterios de diseño de problemas, destinados a mejorar el proceso de enseñanza/aprendizaje en programación, obtuvo resultados donde muestran que, los mecanismos incorporados a la plataforma y su modo de aplicación pueden mejorar la calidad del proceso de enseñanza/aprendizaje. La modalidad de uso utilizada permite garantizar que todos los alumnos tienen un entrenamiento similar que nivela las habilidades para programar, aumenta el interés por la programación y mejora su autoeficacia.

Muchos docentes creen útil solamente en presentar un ejemplo en la computadora para luego explicar su utilidad y que el estudiante trate de comprender la funcionalidad del programa propuesto, han hecho de esta práctica una forma de enseñar a programar sin embargo, estudios y resultados realizados en este campo, mencionan que el trabajo que se haga con el ejemplo y su posterior generalización podrían interpretarse como una predisposición del estudiante hacia un tipo de estrategia en particular. Según un estudio realizado por Miños, A. (2016), en relación a la técnica del ejemplo en pizarra, el 77% de los estudiantes consideró beneficioso comenzar el trabajo a partir de un contenido presentando y explicando un ejemplo del mismo, para luego explicar las características generales del contenido. Menciona además que, el accionar docente influye fuertemente en el hecho de si esto debe considerarse como inducción o no.

El 23% de los estudiantes consideraron que esta estrategia del ejemplo en computadora, no era ni mejor ni peor que comenzar explicando la utilidad del contenido; las respuestas se agruparon como “más o menos”.

Beneficiarios Directos.

La naturaleza de este estudio tiene como sus principales beneficiarios a la comunidad académica quienes tendrán en este proyecto las herramientas necesarias para mejorar el proceso de enseñanza–aprendizaje.

A continuación, los beneficiarios:

- Los profesionales creadores del proyecto.
- Docentes relacionados a informática.
- Alumnos de escuelas y facultades de informática y computación.
- La facultad de Ingeniería de Sistemas.

Beneficiarios Indirectos.

Evidentemente.

- Docentes de otros centros educativos de nivel superior que deseen enriquecer sus métodos y técnicas de enseñanza de la programación.
- Universidades de cualquier parte del mundo que decidan implementar nuevas metodologías de enseñanza de la programación informática a beneficio de los estudiantes dentro de sus Facultades.
- Profesionales independientes que requieren ampliar sus técnicas de programación en base a herramientas de software, métodos entre otros aspectos.

REFERENCIAS BIBLIOGRÁFICAS.

1. Arellano, J., Nieve, O., Arista, G. y Solar A. (2011). Experiencia en el Desarrollo y Uso de un Software para Enseñar Algoritmos. Octavo Congreso Internacional de Computo en Optimizacion y Software. Morelos: México. pp. 265–276.
2. Ball, J. (1989). La micropolítica de la escuela: hacia una teoría de la organización escolar: Paidós Barcelona.

3. Berger, P. L. y Luckmann, T. (1968). La construcción social de la realidad: Amorrortu.
4. Caneo, O. (2016). Las Prácticas Cooperativas como mejora del aprendizaje en la programación de computadores. Repositoria institucional. Universidad de Oviedo. p. 443.
5. Carreño, M., Sandoval, A., Italia, E., Hernandez, J., De Casso, A. y Durán, I. (2015). Entorno web como herramienta de apoyo para el diseño de algoritmos en pseudocódigo durante el proceso enseñanza-aprendizaje. Departamento Academico de Sistemas Computacionales; Univesidad Autonoma de Baja California Sur. pp.296-305.
6. Rosique, F. C., Losilla F. L y Pastor J. F. (2019). Experiencia de aprendizaje activo y colaborativo para la adquisición de competencias en información. España, Universidad Politécnica de Cartagena. Núm. 4, pp. 35-40.
7. Crow, T., Luxton, A. y Wuensche, B. (2018). Intelligent tutoring systems for programming education: a systematic review. in Proceedings of the 20th Australasian Computing Education Conference. Association for Computing Machinery.
8. Del Prado, A. y Lamas, N. (2014). Alternativas para la enseñanza de pseudocódigo y diagrama de Flujos. Revista Electrónica Iberoamericana de Educación en Ciencias y Tecnología. 5(3), p. 102.
9. Estévez, I., Alario, C., Pérez, M., Crespo, R., Leony, D., Parada, A. (2013). Validación por la Comunidad Docente de una Metodología de Aprendizaje Activo para Cursos de Programación. Departamento de Ingeniería Telemática, Universidad Carlos III de Madrid. pp. 563-570.
10. García, N., Santiago, O. y Arellano, J. (2011). Construcción de un esquema de aprendizaje para la enseñanza de algoritmos.
11. Goetz, J. P. y LeCompte, M. D. (1988). Etnografía y diseño cualitativo en investigación educativa. Ediciones Morata.

12. Insuasti, J. (2016). Problemas de enseñanza y aprendizaje de los fundamentos de programación. **10(2)**: pp. 234-246.
13. Lara, J., Andalira, G., Morán, R. S., Jordán, A. E., & Ramos Carpio, J. P. (2018). Estrategias didácticas aplicadas en la formación de competencias. Estudiantes-Ingeniería Comercial-FAFI-Universidad Técnica de Babahoyo. Dilemas Contemporáneos: Educación, Política y Valores. Año: VI, Número: Edición Especial, Artículo no.: 8, Período: Julio, 2018.
<https://dilemascontemporaneoseduccionpoliticayvalores.com/files/200003885-5b1bc5c185/EE%2018.7.08%20Estrategias%20did%C3%A1cticas%20aplicadas%20en%20la%20formaci%C3%B3n%20de.....pdf>
14. López, J., Hernández, C. y Farran, Y. (2011). Una plataforma de evaluación automática con una metodología efectiva para la enseñanza/aprendizaje en programación de computadores. *Ingeniare. Revista Chilena de Ingeniería*, **19(2)**: p. 265-277.
15. Lovos, E., Gonzalez, A. y Bertone, R. (2013). Experiencia de utilización de Herramientas Colaborativas para la enseñanza y el aprendizaje de la Programación de Computadoras. XVIII Congreso Argentino de Ciencias de La Computación.
16. Miños, A., (2016). Uso didáctico de estrategias inductivas en un curso introductorio de programación estructurada: pp. 95-110.
17. Muñoz, R., Barcelos, S., Villarroel, R., Barría, M., Becerra, C., Noel, R. y Frango, I. (2015). Uso de Scratch y Lego Mindstorms como Apoyo a la Docencia en Fundamentos de Programación. Chile, Universidad de Valparaíso. pp.248-254.
18. Ocegüera, S., Expósito, C., Díaz, G. y Bonne, E. (2011). Metodología de la enseñanza de la informática. Editorial EDUCACION CUBANA.
19. Popkewitz, T. S., (2010). Paradigmas e ideología en la investigación educativa: documento de lectura: International Alliance for Learning.

20. Sánchez, G. T., Aurea, B., Carriel, D. & Ramos, J. P. (2019). La Internacionalización de la Educación Superior, desde la perspectiva educacional, cinco retos y una necesaria reflexión sobre el sentido del proceso. *Dilemas Contemporáneos, Educación, Política y Valores*. Año: VI, Número: Edición Especial, Artículo no.:4, Período: Junio, 2019.
- <https://dilemascontemporaneoseduccionpoliticyvalores.com/files/200005025-9d9119e8f0/EE%2019.06.04%20La%20Internacionalizaci%C3%B3n%20de%20la%20Educaci%C3%B3n%20Superior..pdf>
21. Spigariol, L. (2015). Recursos pedagógicos tecnológicos para aprender a programar. Universidad Tecnológica Nacional, Facultad Regional Buenos Aires.
22. Sierra, F. S., (2006). *Métodos de investigación etnográfica en Ciencias Sociales*: Mira.
23. Trejos, O. (2013). Los articulos científicos como metodologia de aprendizaje en el 1º semestre de un programa de Ingeniería. *Scientia et Technica*. **18** (3): pp. 530-534.
24. Timarán, R., Toledo, J. y Chávez, A. (2012). Programación multiparadigma como estrategia de aprendizaje en los lenguajes de programación en ingeniería en sistemas. Núm. 27.
25. Taylor, S. J., Bogdan, R. y Piatigorsky, J. (1987). *Introducción a los métodos cualitativos de investigación: la búsqueda de significados*: Paidós.
26. Vidal, C., Cabezas, C., Parra, J. y López, L. (2015). Experiencias Prácticas con el Uso del Lenguaje de Programación Scratch para Desarrollar el Pensamiento Algorítmico de Estudiantes en Chile. *Formación Universitaria*. **8** (4): pp. 23-32.
27. Zuleta, A. y Chávez, A. (2011). Uso de herramientas informáticas como estrategia para la enseñanza de la programación de computadores. Núm. 57, pp. 23-32.

DATOS DE LOS AUTORES

- 1. Rosa Elizabeth Molina Izurieta.** Magister en Administración y Dirección de Empresas. Docente de la Universidad de Guayaquil, Guayas, Ecuador, email: rositaelizabeth28@hotmail.es
- 2. Renzo Rogelio Padilla Gómez.** Magister en Docencia y Gerencia en Educación Superior. Docente de la Universidad de Guayaquil, Guayas, Ecuador, email: renzo.padillag@ug.edu.ec
- 3. Maikel Yelandi Leyva Vázquez.** Doctor en Ciencias Técnicas. Docente de la Universidad de la Universidad Politécnica Salesiana. Guayaquil, Ecuador, Instituto Superior Tecnológico Bolivariano de Tecnología, Guayaquil, Ecuador, email: mleyvaz@gmail.com

RECIBIDO: 11 de octubre del 2019.

APROBADO: 20 de octubre del 2019.