



uaim

RA XIMHAI

Volumen 11 Número 4 Edición Especial
Julio - Diciembre 2015
289-304

ANÁLISIS DE LOS PROBLEMAS DE APRENDIZAJE DE LA PROGRAMACIÓN ORIENTADA A OBJETOS

ANALYSIS OF LEARNING PROBLEMS OF OBJECT-ORIENTED PROGRAMMING

José Emilio **Sánchez-García**¹; Margarita **Urías-Ruiz**² y Brenda Edith **Gutiérrez-Herrera**³

¹Profesor-Investigador de la Universidad Autónoma Indígena de México, Los Mochis, Ahome, Sinaloa, México. ²Jefe de Carrera de Ingeniería en Sistemas Computacionales de la Universidad Autónoma Indígena de México, Mochicahui, El Fuerte, Sinaloa, México. ³Facilitador Educativo de la Universidad Autónoma Indígena de México, Los Mochis, Ahome, Sinaloa, México. Responsable: José Emilio Sánchez García. Fuente de Cristal 2334 entre Coral y Cuarzo. Fraccionamiento Fuentes del Bosque. C.P. 81229. Los Mochis, Ahome, Sinaloa, teléfono 01 (668) 8160-320 al 350. Correo electrónico emilio_507@hotmail.com.

RESUMEN

Los problemas en el aprendizaje de la programación orientada a objetos son diversos y cobra relevancia investigarlos debido a que independientemente de la generación de estudiantes siempre se presenta como constante el problema de la comprensión y aplicación de conceptos para la resolución de problemas de automatización, expresados en líneas de código de algún lenguaje de programación, para abordarlos se ha recurrido al paradigma cualitativo con el método etnográfico utilizando la estrategia de observación participante y la entrevista en profundidad a maestros y semi-estructurada a estudiantes, como resultado se obtuvo que la frecuencia más alta fue la falta de disciplina mental en el manejo de la lógica por parte de los estudiantes al momento de programar, concluyendo que para desarrollar esta lógica es necesario una asignatura previa de metodología de desarrollo de software y de modelos utilizando la notación UML.

Palabras clave: cambio de paradigma, entorno, hardware, alumnos.

SUMMARY

Problems in learning object-oriented programming are diverse and investigate becomes relevant because regardless of the generation of students is always presented as the constant problem of understanding and application of concepts to solve problems of automation, expressed in lines of code in some programming language to address them has resorted to qualitative paradigm with the ethnographic method using the strategy of participant observation and in-depth interview teachers and semi - structured for students as a result it was found that the highest frequency It was the lack of mental discipline in handling logic by students when programming , concluding that to develop this logic a prior course of software development methodology and models using the UML notation is necessary.

Key words: paradigm shift, environment, hardware, students.

INTRODUCCIÓN

La Universidad Autónoma Indígena de México es una institución intercultural del estado de Sinaloa que tiene como misión propiciar la formación integral de profesionales competentes, críticos y comprometidos con el desarrollo sustentable de las comunidades. Cuenta dentro de su oferta educativa con la carrera de Ingeniería en Sistemas Computacionales con énfasis en software, la

cual es muy demandada por los estudiantes. Una de las asignaturas dentro de este programa educativo es la programación orientada a objetos.

La enseñanza de la materia de programación orientada a objetos en la UAIM ha sido una tarea desempeñada en los últimos siete años, sin embargo esta experiencia indica que no queda muy claro ¿cómo resolver el problema del aprendizaje de la programación orientada a objetos?

El problema del aprendizaje de la programación orientada a objetos se manifiesta toda vez que es una materia compleja que implica la integración de muchos elementos como son el paradigma orientado a objetos, el lenguaje de programación, el entorno de desarrollo, la metodología de desarrollo, el lenguaje de modelado, los patrones de desarrollo y la lógica de programación. Por lo tanto los alumnos se encuentran ante una cantidad abrumadora de conceptos en un periodo corto de tiempo, lo que dificulta su asimilación y el desarrollo de las habilidades para generar líneas de código como lo explica Spigariol y Passerini (2013):

Los docentes veían en los estudiantes que el uso del lenguaje representaba una curva de aprendizaje abrupta en los primeros momentos de la materia ya que requieren el manejo de una cantidad amplia de conceptos antes de poder realizar algo relativamente sencillo (...). La disociación entre teoría y práctica que se generaba era ciertamente contraproducente y dificultaba el proceso de aprendizaje.

En este mismo sentido Brooks (1987) explica que:

(...) Software entities are more complex for their size than perhaps any other human construct because no two parts are alike (at least above the statement level) (...). The essence of a software entity is a construct of interlocking concepts: data sets, relationships among data items, algorithms, and invocations of functions. This essence is abstract in that such a conceptual construct is the same under many different representations. It is nonetheless highly precise and richly detailed.

Debido a que los conceptos que se tratan en la asignatura de programación orientada a objetos son muchos y en algunos casos difíciles de comprender por el alto nivel de abstracción, esto se configura como un factor que dificulta el aprendizaje, esto se ha visto reflejado por ejemplo cuando los estudiantes generalmente no distinguen entre lo que es una clase y un objeto.

El problema del aprendizaje de la programación orientada a objetos se aborda porque la academia de ingeniería en sistemas computacionales con énfasis en software de la Universidad Autónoma Indígena de México (UAIM) ha observado que independientemente de la generación de estudiantes, siempre existe el problema de la comprensión y aplicación de conceptos para la resolución de problemas de automatización, expresados en líneas de código de algún lenguaje de programación como la explican Ramírez, M., Lucio, M., Garza, J., García, L. y Vargas, J. (2011) *regularmente se ha encontrado resistencia de los alumnos para aprender programación en los lenguajes utilizados.*

Está claro que si no se analiza y atiende este problema del aprendizaje de la programación orientada a objetos persistirá el hecho de que solo uno cuantos estudiantes de cada generación realmente comprenden y aplican los conceptos propios de la materia, de manera que se reduce

las posibilidades de que la mayoría de los estudiantes en ingeniería en sistemas computacionales con énfasis en software y base de datos culminen sus estudios de licenciatura.

En el caso de la Universidad Autónoma Indígena de México (UAIM) el problema surge en gran medida en virtud de que el plan de estudios actual no se encuentra consideradas las asignaturas de estructura de datos y algoritmos, previo a la enseñanza de la programación de computadoras. Estas materias son necesarias para desarrollar en el estudiante la capacidad de razonamiento lógico para resolver problemas mediante computadoras.

Ahora bien los estudiantes inician la programación en computadoras con el paradigma estructurado, el cual consiste en la utilización de subrutinas y tres estructuras de control únicamente: secuencia, selección e iteración motivo por el cual es necesario un proceso de transición hacia el paradigma orientado a objetos, en el cual se utilizan los objetos para la interacción a partir de la definición de clases o categorías, tratando de modelar la realidad constituyéndose en una forma diferente de programar, esta situación se configura como un factor que dificulta en gran medida el aprendizaje de la programación orientada a objetos.

Además la forma de enseñar la asignatura de programación orientada a objetos es muy similar a la de la programación estructurada, primero se tratan temas básicos del lenguaje de programación, como son la declaraciones de los tipos de datos, las estructuras de control y las sentencias de condición, para posteriormente enseñar lo que son las clases, objetos y los principales temas propios del paradigma orientada a objetos, esto contribuye a que los estudiantes sientan que continúan con el mismo paradigma de programación estructurada.

En virtud de que los estudiantes no logran asimilar este cambio de paradigmas, usan el lenguaje de programación orientada a objetos como un lenguaje de programación estructurada, por consiguiente no resuelven problemas diseñando clases.

Así pues los estudiantes al llegar a la asignatura de programación orientada a objetos se enfrentan de entrada a dos problemas, la percepción que tienen de dificultad y por otro lado el proceso de transición del paradigma estructurado hacia el paradigma orientado a objetos.

Por lo tanto los docentes deberían planificar su función pensando en nuevos métodos, estrategias y actividades para facilitar el aprendizaje, como puede ser buscar nuevas herramientas de *software* que proporcionen el andamio necesario para el aprendizaje de los estudiantes que apenas comienzan.

Una de las dificultades para el aprendizaje de la programación orientada a objetos lo constituyen las constantes actualizaciones de los entornos de programación por los avances de la misma tecnología y por otro lado los lenguajes se actualizan también constantemente dejando algunas sentencias obsoletas y agregando muchas otras.

Puesto que los entornos de programación son diseñados para su uso profesional, cuentan con una gran cantidad de herramientas y opciones de interfaz gráfica que resultan ser abrumadoras para los estudiantes que apenas inician, esto se constituye en otro factor que dificulta el aprendizaje de la programación orientada a objetos, por lo tanto existe necesidad de entornos de programación diseñados específicamente para la aprendizaje de los principiantes.

Un factor importante que dificulta el aprendizaje de la programación orientada a objetos lo constituyen las exigencias de sintaxis del lenguaje de programación, en ocasiones los estudiantes después de un periodo de concentración han logrado comprender los conceptos, sin embargo cuando editan los programas en el entorno de desarrollo, por un solo carácter todo un programa deja de funcionar lo que ocasiona frustración en los estudiantes.

Una característica observada con relación con las exigencias de las reglas de escritura, es que los estudiantes al editar el código en el entorno de desarrollo, el compilador muestra una gran cantidad de errores ante lo cual los estudiantes sienten que no saben programar, cuando en realidad la función del compilador es identificar los errores de escritura para apoyar al programador en su solución.

Además dado a que existe una diversidad de lenguajes de programación para el aprendizaje de ambos paradigmas, esto contribuye a un aumento en la complejidad tanto para los docentes como para el estudiante, en virtud de que los diferentes lenguajes tienen sus propias sentencias y reglas de escritura.

Los actores involucrados en esta situación son los docentes y estudiantes del área de Ingeniería en Sistemas Computacionales con énfasis en Ingeniería en *Software*, sin embargo el mismo problema del aprendizaje de la programación se extiende hacia el énfasis en Telecomunicaciones debido a que llevan asignaturas como la de micro-controladores donde es necesario el desarrollo de competencias para la generación de código.

Cuando el docente enseña la asignatura de programación orientada a objetos en el centro de cómputo, se presentan casos específicos de estudiantes que se retrasan en un punto de la codificación, esto se debe a que existen distintas velocidades para aprender, diversidad en las habilidades, así como diferentes actitudes como lo explica Sousa (2002) *Como la velocidad de aprendizaje y la de recuperación son independientes, los individuos pueden aprender rápido o lentamente, recuperar datos rápido o lentamente o mostrar cualquier combinación de dichas capacidades.*

Así pues, si el docente se detiene con un estudiante retrasado, entonces los estudiantes con mayores velocidades se tienen que esperar, si por el contrario se les indica que sigan avanzando mientras se atienden a los atrasados, entonces se tiene un grupo heterogéneo con distintos niveles de avance, lo que hace difícil la enseñanza grupal puesto esto requiere de una atención personalizada para atender las distintas dudas que se originan como resultado de sus diferentes niveles de avance autónomo, que por cuestiones de tiempo es imposible de atender en una sola sesión de hora y media a la semana por asignatura como la establece la normatividad de la universidad.

La solución que se ha implementado es solicitar a estos estudiantes avanzados que apoyen a sus compañeros atrasados, lo que les permite consolidar su conocimiento y también colaborar con otros para construirlo. Sin embargo existe la posibilidad de que en un grupo dado, se cuente con estudiantes con buen ritmo de aprendizaje pero que no compartirán con los demás su conocimiento. Así mismo los estudiantes atrasados esperarían que los alumnos avanzados los apoyen sin realizar ningún esfuerzo por aprender de forma autónoma.

Ahora bien los docentes que enseñan la asignatura de programación orientada a objetos son profesionistas, uno relacionado con los sistemas computacionales y el otro en informática, que no cuentan con formación en educación, lo cual se configura como un factor determinante que afecta el proceso de enseñanza aprendizaje tal como lo señala el departamento de desarrollo académico de la UAIM (2011), (...) *la formación pedagógica-didáctica del cuerpo docente de la UAIM detectados como problemáticos en el diagnóstico realizado.*

En virtud de que los conocimientos en programación orientada a objetos son los cimientos para comprender otras materias a lo largo del plan de estudios, las repercusiones que puede tener el problema del aprendizaje de la programación, no son cosa menor, trascienden incluso la institución educativa, cuando los egresados se encuentren en el ámbito profesional deberán contar con las competencias en el manejo del paradigma orientado a objetos, este paradigma es el de mayor uso en la actualidad, desde páginas *Web*, computadoras de escritorios, *software* incrustado en tarjetas hasta dispositivos móviles trabajan de esta forma.

Los tipos de efectos que provoca las dificultades en el aprendizaje de la programación orientada a objetos, son el incremento en los indicadores de reprobación y deserción, por lo tanto para evitar esto algunos estudiantes se cambian al énfasis de telecomunicaciones o bien a otras carreras.

Los principales afectados por las dificultades en el aprendizaje de la programación orientada a objetos son los estudiantes, debido a que no logran desarrollar las competencias necesarias para desempeñarse en el ámbito profesional como programadores de sistemas, analistas o líderes de proyectos de desarrollo de *software*.

Ahora bien como resultado de las dificultades en el aprendizaje de la programación orientada a objetos, también se ve afectada la imagen de la institución educativa toda vez que el reconocimiento social se debe en gran medida a la calidad que tengan sus estudiantes y egresados en relación con sus competencias.

Si se considera los problemas antes mencionados en el aprendizaje de la programación orientada a objetos, se puede apuntar que no se hecho nada para solucionarlos, no obstante se han realizado algunos cambios académicos importantes resultado del informe de la fase de diagnóstico elaborado por el departamento de desarrollo académico de la UAIM (2011), donde se generó información para fundamentar la propuesta de intervención en la universidad como es el primero de los cuatro ejes denominado *I.-Eje de formación pedagógica didáctica para la profesionalización del cuerpo docente de la UAIM*, en consecuencia se inició un diplomado en docencia universitaria intercultural, lo cual será un factor que favorece el proceso de enseñanza aprendizaje de la asignatura, debido a que los actuales docentes son profesionistas en el área de computación y no cuentan con formación en educación.

Recientemente se rediseño el plan de estudios de trimestres a semestres donde se contempla la asignatura de algoritmos y fundamentos de programación, los cuales son necesarias para la formación previa de los estudiantes en la resolución de problemas de computadoras.

El problema del aprendizaje de la programación orientada a objetos lo ha abordado la academia de ingeniería en sistemas computacionales formado por los docentes del énfasis en *software* y bases de datos, donde se han analizado los distintos problemas que se presentan tanto en la práctica docente como en el aprendizaje de la materia, para tratar de solucionarlos. No obstante

las acciones que se han llevado a cabo son pocas en relación con la complejidad del problema, además que los docentes no cuentan con formación en educación como para tratar de resolverlo desde el punto de vista pedagógico.

Como se puede observar el problema del aprendizaje de la programación orientada a objetos es complejo, toda vez que son diversas las causas que lo originan así como también los factores que la provocan, sería muy ambicioso el tratar de resolver todo el problema en su conjunto, sin embargo con el apoyo de las tecnologías de la información y comunicación es factible resolver algunos asuntos.

El abordar el aprendizaje de la programación orientada a objetos como tema de investigación es importante, toda vez que intenta resolver un problema que tiene consecuencias económicas y sociales. La nueva herramienta de trabajo es el conocimiento, un profesionista en software requiere conocer y aplicar conceptos para generar tecnología.

Objetivo de Investigación

Analizar los problemas en el proceso de aprendizaje de la programación orientada a objetos en los estudiantes de la generación 2012-2016 de la carrera de ingeniería en sistemas computacionales de la UAIM.

MATERIALES Y MÉTODOS

La presente investigación se desarrolló en la Universidad Autónoma Indígena de México al interior de cuatro grupos de sexto trimestre de Ingeniería en Sistemas Computacionales, en la asignatura de programación orientada a objetos en las Unidad Los Mochis y la Unidad Mochicahui. El estudio se realizó a todo el universo de estudiantes dado que los grupos son pequeños, correspondiendo tres grupos a la Unidad Mochicahui y un grupo en la Unidad Los Mochis.

Los grupos pertenecen a la generación 2012-2016 y está conformado por 12 integrantes, nueve hombres y tres mujeres cuyas edades varían entre 19 y 25 años en el caso de la Unidad Los Mochis. En la Unidad Mochicahui son tres grupos integrado por 15, 16 y 17 estudiantes respectivamente, el primer grupo está integrado por 12 hombres y tres mujeres, el segundo son 12 hombres y cuatro mujeres, el tercer grupo está formado por 13 hombres y cuatro mujeres. Las edades varían entre 19 y 22 años en el primer grupo, el segundo grupo entre 19 y 25 y el tercer grupo entre 19 y 23 años tal como se muestra en la *Figura 1*. Los estudiantes son originarios de Mochicahui, El Fuerte, Sinaloa y de poblados cercanos como el 2 de abril, 5 de mayo, Constanza, Santa Cruz, El Carrizo y Jahuara I.

Algunos padres de los estudiantes de la Unidad Mochicahui hablan dos idiomas (Mayo-Yoreme - Español), es decir pertenecen a familias que tienen como lengua materna el Mayo-Yoreme y que aprendieron como segunda lengua el Español. Se trata de alumnos que proceden de familias del sector indígena rural que tienen sus creencias y costumbres arraigadas y son de escasos recursos económicos. Algunos estudiantes participan como judíos, músicos y en la danza del venado en las festividades religiosas y rituales de sus centro ceremoniales.



Figura 1.- Estudiantes de la Unidad Mochicahui.

La institución se encuentra ubicada en Fuentes de Cristal 2334 entre Coral y Cuarzo, Fraccionamiento Fuentes del Bosque, C.P. 81290, en Los Mochis, Ahome, Sinaloa y cuenta con todos los servicios ya que se localiza en un contexto urbano, ofrece las licenciaturas de Derecho, Psicología Social Comunitaria, Sociología Rural, Contaduría, Ingeniería en Sistemas Computacionales e Ingeniería en Sistemas de Calidad, en el caso de la unidad Los Mochis y en el caso de la Unidad Mochicahui se encuentra ubicada en Juárez N° 39, localizada en zona rural y tiene una oferta educativa de nueve programas educativos.

En el caso de la Unidad Los Mochis la asignatura se impartió en el en el centro de cómputo él cual se encuentra ubicado en planta baja, cuenta con un espacio de 42 metros cuadrados (seis por siete metros), equipo de cómputo consistente en 18 computadoras de escritorio marca LENOVO color negro con procesador Intel Core 2 Duo CPU E 8400 de 3,000 GHZ, memoria RAM de 2 Gb, disco duro de 250 Gb de capacidad, monitor LCD pantalla plana y ratón óptico. Un video-proyector BenQ, SVGA 800 X 600 de 3000 lúmenes color negro. Un concentrador de 24 puertos marca 3Com y un modem para el acceso a internet tal como se muestra en la *Figura 2*.



Figura 2- Centro de computo Unidad Los Mochis.

Paradigma de Investigación

El enfoque de la presente investigación es cualitativo. Se apoyó en el método etnográfico para comprender las perspectivas de los diferentes actores involucrados en este caso estudiantes y maestros, por lo tanto se inscribe en el paradigma interpretativo y constructivista. Resulta necesario graficar la secuencia de recolección, análisis e interpretación de los datos, como se muestra en la *Figura 3*.

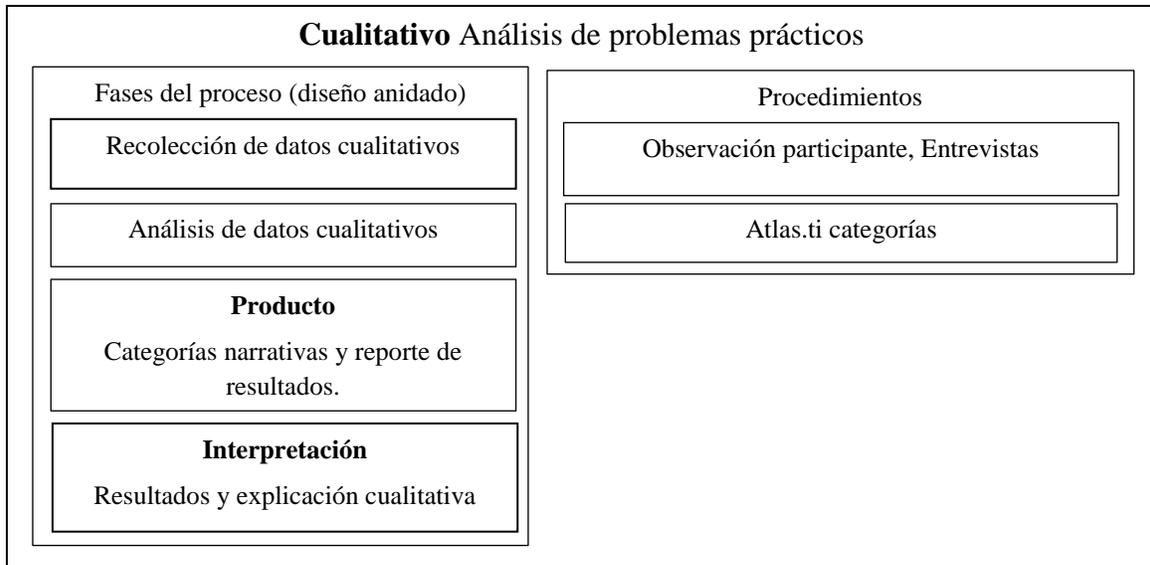


Figura 3.- Visualización de la recolección, análisis e interpretación del estudio.

Método de investigación

Se consideró el objetivo como criterio para seleccionar el método de investigación:

Analizar los problemas en el proceso de aprendizaje de la programación orientada a objetos en los estudiantes de la generación 2012-2016 de la carrera de ingeniería en sistemas computacionales de la UAIM.

Por lo tanto se seleccionó el método etnográfico toda vez que fue necesario conocer en el contexto cotidiano el proceso de aprendizaje, se utilizó la estrategia de observación participante y la entrevista. A continuación se describe en detalle la forma que se llevó a cabo este diseño de investigación.

Análisis de problemas prácticos

Se realizó la observación participante para conocer en su ámbito natural cuales son los problemas en el proceso de aprendizaje de los estudiantes en la programación orientada a objetos. Se realizaron anotaciones en un diario de campo y se utilizó una grabadora para contar con elementos para su registro y análisis posterior, esto apoyó en la identificación patrones y la construcción de categorías. En este proceso de análisis de datos se utilizó como apoyo la herramienta *Atlas.ti*.

La observación participante se realizó en tres grupos de la Unidad Mochicahui y un grupo de la Unidad Los Mochis. Así mismo se complementó la estrategia de observación participante con entrevistas en profundidad a maestros y a los alumnos se les realizaron entrevistas semiestructuradas con el objetivo de conocer las perspectivas y percepciones sobre las dificultades en el proceso de aprendizaje de la programación orientada a objetos.

Específicamente se entrevistó un maestro de la Unidad Mochicahui y un maestro de la Unidad Los Mochis. Respecto a los estudiantes se entrevistaron a 15, 16 y 17 estudiantes correspondientes a tres grupos de la Unidad Mochicahui y 12 estudiantes del grupo único de la Unidad Los Mochis.

Posteriormente las entrevistas en profundidad y semiestructuradas se transcribieron y su contenido fue analizado inductivamente para construir categorías mediante la comparación constante de unidades de análisis utilizando como apoyo la herramienta *Atlas.ti* sobre la base de las recurrencias observadas en los testimonios de los alumnos y los profesores. Finalmente se integraron las conclusiones y resultados.

De acuerdo con Rodríguez y Valdeoriola (s.f.), *las estrategias o técnicas se refieren a modos, maneras o estilos de recoger la información, mientras que los instrumentos son herramientas concretas de cada técnica o estrategia que nos permiten llevar a la práctica la obtención de la información.*

En este sentido la presente investigación de carácter cualitativo con un diseño de investigación apoyado con el método etnográfico emplea diferentes estrategias o técnicas de recolección de información como es la observación participante, la entrevista en profundidad a maestros y entrevistas semiestructuradas a alumnos con la intención de describir las perspectivas y percepciones de los problemas en el proceso de aprendizaje de la programación orientada a objetos así como de diferentes instrumentos como diario de campo, guía de entrevistas y cuestionarios estos últimos complementan y enriquecen la información cualitativa, la figura 4 muestra cómo quedarían integrado el método y las técnicas.

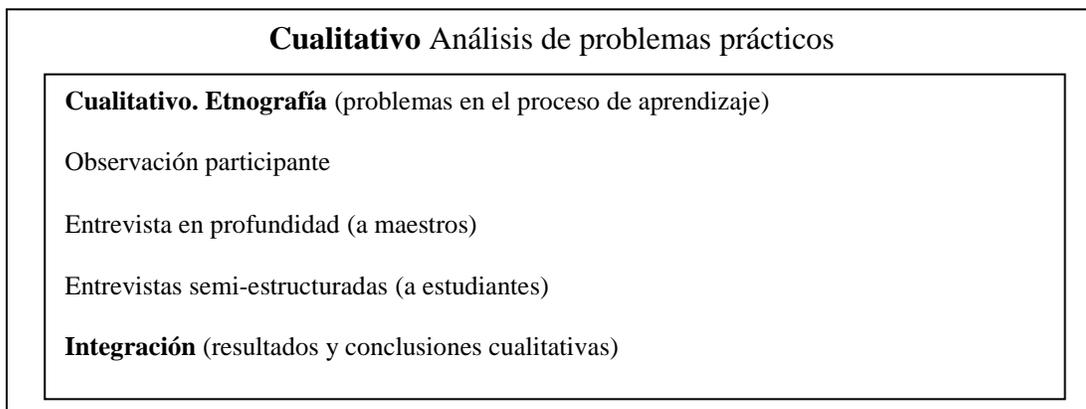


Figura 4.- Integración del método y las técnicas cualitativas.

Instrumentos para la recolección de datos

En la presente investigación de carácter cualitativo se han empleado los siguientes instrumentos:

1. El diario de campo donde se realizaron anotaciones de la observación participante para conocer en su ámbito natural cuáles son los problemas en el proceso de aprendizaje de la programación orientada a objetos. El diario de campo es un instrumento que permite llevar a la práctica la observación participante.
2. Guía de entrevista en profundidad a maestros para conocer las perspectivas y percepciones sobre las dificultades en el proceso de aprendizaje de la programación orientada a objetos.

3. Guía de entrevista semiestructurada a los alumnos para conocer las perspectivas y percepciones sobre las dificultades en el proceso de aprendizaje de la programación orientada a objetos.

Se aplicó un instrumento de recopilación de información a estudiantes denominado “Guía de entrevista a estudiantes para conocer las dificultades en el aprendizaje de la programación orientada a objetos”. Así mismo se aplicó un instrumento para recabar información a maestros denominado “Guía de entrevistas a maestros para conocer las dificultades en el aprendizaje de la programación orientada a objetos”.

Se transcribieron en forma de texto las anotaciones del diario de campo así como 62 entrevistas generándose 140 hojas en Microsoft Word, de las cuales correspondían a 13 entrevistas de la unidad académica de Los Mochis y 49 de la Unidad Mochicahui y se incorporaron como documentos primarios en la herramienta *Atlas.ti* para formar categorías sobre la base de las recurrencias y también se formaron relaciones entre conceptos.

Para el caso de la unidad académica de Los Mochis se analizaron 12 entrevistas aplicadas a estudiantes y una entrevista a una maestra en relación con las perspectivas y percepciones de las dificultades en el aprendizaje de la programación orientada a objetos con el apoyo de la herramienta *Atlas.ti* donde la unidad de análisis fueron las intervenciones de los participantes. En la unidad Mochicahui se analizaron 48 entrevistas aplicadas a estudiantes y una entrevista a un maestro.

RESULTADOS Y DISCUSIÓN

En este apartado se presentan los resultados sobre los problemas presentados en el aprendizaje de la programación orientada a objetos de los cuales emergieron los siguientes temas: cambio de paradigma estructurado al paradigma orientado a objetos, entornos de programación, hardware y alumnos.

Cambio del paradigma estructurado al paradigma orientado a objetos

Este tema se refiere a que el cambio entre el paradigma estructurado al paradigma orientada a objetos es un cambio radical, en el primero el problema se va adaptando en un primer momento a un pseudocódigo para después transformarlo en un diagrama de flujo, finalmente este diagrama se convierte en código de algún lenguaje de programación estructurada como puede ser *Pascal*, es decir el problema se adapta al lenguaje, en cambio en el paradigma orientada a objetos el programa se adapta a los problemas de la vida real, los estudiantes inician con el paradigma estructurado en la materia anterior de tal forma que cuando inician la programación orientada a objetos siguen pensando en esta forma lo que se configura en otro factor que dificulta el aprendizaje de la asignatura.

Sobre el tema cambios del paradigma estructurado al paradigma orientado a objetos surgieron tres categorías relacionadas: La forma de presentar la teoría, nivel de abstracción de los conceptos y la cantidad de carga teórica.

CATEGORÍA. Forma de presentar la teoría.

no siento que sea en si la teoría lo que sea un obstáculo sino la manera en que está planteada la manera en que se enseña pienso que es muy difícil de comprender pero es necesaria siento que si no la tuviera seria todavía más difícil de comprender pero habría que hacerla más accesible o más entendible (Estudiante).

Los estudiantes manifestaron que la forma de plantear la teoría en esta materia es lo que la hace difícil de comprender, esto se interpreta que es necesaria la transformación de la teoría como se presenta en los libros en otra forma más sencilla que se encuentre al nivel de los estudiantes que apenas inician, o quizás utilizando un lenguaje cercano al contexto, sucede que la mayoría de los libros de programación orientada a objetos son traducciones de textos en inglés o son de otros países de habla hispana como España y donde definen los términos de acuerdo a su contexto. Sin embargo esto implicaría también cambiar las herramientas de desarrollo integrado que también son hechas en el extranjero y se traducen al español, por lo tanto lo que hace falta es una solución donde se transforme de forma integral no solo los textos, sino también las herramientas de apoyo en la generación de código.

CATEGORÍA. Nivel de abstracción de los conceptos.

Allí está un poquito más complejo no porque como de un inicio no no no responden o no alcanzan a captar porque las cosas se van ordenando por clases objetos (Maestro).

Esta categoría significa que los conceptos son difíciles de entender por el alto nivel de abstracción motivo por el cual se configura como barrera para su aprendizaje.

CATEGORÍA. Cantidad de carga teórica.

Me tocó a mí dar la materia anterior a los que ellas llevan de programación orientada a objetos y en la materia anterior igual está un poco cargado, este yo creo que no porque debería dividirse esa materia en dos me imagino por el tiempo, por la carga de conceptos dividirse en dos materias (Maestro).

De acuerdo con la entrevista de la maestra la cantidad de carga teórica es un factor que afecta el aprendizaje toda vez que son muchos los conceptos que se ven en solo doce sesiones al trimestre. En el UAIM no existen las clases diarias, solo talleres de hora y media a la semana por cada asignatura. Por lo tanto se configura como otros de los factores que afectan el aprendizaje de los estudiantes.

Entornos de programación

Sobre el tema entornos de programación surgieron tres categorías relacionadas: Complejidad del entorno integrado de desarrollo, lenguaje de programación en inglés y mensajes del compilador en inglés.

CATEGORÍA. Complejidad del entorno integrado de desarrollo.

Me parece un poco avanzado el entorno, muchos términos o plantillas no conocemos no sé si con el tiempo nos la vayan a facilitar o enseñar, si hay algunas cosas raras que se ven en el entorno (Estudiante).

Esta categoría se refiere a que el entorno de desarrollo integrado está diseñado para su uso profesional y no con fines didácticos, la cantidad de herramientas y opciones abruma al estudiante que apenas inician en la programación orientada a objetos configurándose como un elemento que dificulta el aprendizaje.

CATEGORÍA. Lenguaje de programación en inglés.

Si tuviera en español estaría más fácil entiendo en español porque en inglés es muy poco lo que se entiende (Estudiante).

En esta categoría los estudiantes manifestaron que el hecho de que el lenguaje de programación orientada a objetos se encuentre en inglés se constituye en otro factor que afecta el aprendizaje toda vez que les dificulta la comprensión de las sentencias. Esto cobra mayor relevancia en el contexto de la unidad Mochicahui donde los estudiantes proceden de comunidades indígenas del país cada uno con sus propias lenguas donde el español es su segunda lengua y el inglés se constituye en la tercera por lo tanto en algunos casos es más difícil el entendimiento de las instrucciones del lenguaje de programación orientada a objetos en inglés.

CATEGORÍA. Mensajes del compilador en inglés.

Es algo muy gracioso porque el compilador yo les digo es que casi te habla allí les dice la línea, les dice el error a veces es un simple punto y coma pero el hecho allí si yo creo se ciclan no al ver texto en inglés así como que haaay no sé lo que dice (Maestra).

La función de compilador es el de ayudar al programador a encontrar los errores de sintaxis, indicando el tipo de error y la línea donde se encuentra, incluso a este mensaje se les puede hacer un clic y lleva exactamente en el punto donde se ubica, sin embargo de acuerdo a lo que manifestó la maestra los mensajes se presentan en inglés y a los estudiantes les cuesta leerlo, aunado a el hecho de que no les gusta leer y que los errores los consideran como algo que los desmotiva.

Hardware

Sobre el tema hardware surgieron dos categorías relacionadas: Falta de equipo de cómputo y el centro de cómputo funciona como salón de clases al mismo tiempo.

CATEGORÍA. Falta de equipos de cómputo.

A veces también batallamos de que no hay máquinas porque están otros alumnos (Estudiante).

En ocasiones algunos estudiantes no pueden utilizar los equipos de cómputo debido a que se encuentran ocupados, esto se debe a que son insuficientes para la cantidad de estudiantes que se

encuentran en la Unidad Los Mochis, por lo tanto se constituye como otro factor que afecta el aprendizaje debido a que no pueden realizar sus prácticas fuera de las sesiones de talleres programados.

CATEGORÍA. Centro de cómputo funciona como salón de clases al mismo tiempo.

Desgraciadamente aquí en la universidad el centro de cómputo es uno, está muy pequeño allí dan clases yo pienso que debería haber un centro de cómputo, especialmente para el alumno que vaya allí hacer tareas trabajos prácticas porque en ocasiones incluso para todos es incómodo para el maestro que está dando la sesión allí el taller y que alumno quiera entrar a veces si hay espacios pero también está interrumpiendo entonces yo creo que debería a ver un área exclusiva para que alumno haga desde tareas de cualquier materia hasta también para programar (Maestro).

Esta categoría se refiere al hecho que en el caso de la UAIM unidad Los Mochis existe un solo centro de cómputo que funciona al mismo tiempo como sala de aprendizaje de los programas educativos que se imparten motivo por el cual siempre está ocupado y los estudiantes no pueden realizar prácticas extra-clase de la asignatura además que no existe privacidad para impartir la materia y tampoco para funcionar como centro de cómputo.

Alumnos

Sobre el tema alumnos surgieron tres categorías relacionadas: La no atracción hacia la programación, falta de disciplina mental en el manejo de la lógica y a los estudiantes no les gusta leer.

CATEGORÍA. No atracción hacia la programación.

Pues yo creo que es un poco complicada, sin embargo sí se puede hacer, ahorita no me atrae mucho, me atraen un poquito más las telecomunicaciones (Estudiante).

Esta categoría podría interpretarse que el hecho de que no les atraiga la programación de computadoras repercute en una escasa motivación por lo tanto se presenta como algo que afecta su aprendizaje.

CATEGORÍA. Falta de disciplina mental en el manejo de la lógica.

O.k. Buenos días este yo pienso que un poco de la problemática de lo que es la programación orientada a objetos es la lógica yo creo que para entrar a lo que es la programación anteriormente deben llevar alguna materia donde ellos puedan razonar antes de conocer el lenguaje razonar un poco lo que es la lógica y de cómo se trabaja, que aprendan a pensar orientada a objetos (juju) (Maestro).

Esta categoría se refiere que los estudiantes no han desarrollado el pensamiento lógico matemático motivo por el cual se les dificulta el aprendizaje de la asignatura de acuerdo a lo que manifestó la maestra en su experiencia como docente.

CATEGORÍA: A los estudiantes no les gusta leer.

Yo creo que a los muchachos a veces como que no les gusta mucho leer (Maestro).

Esta categoría nos indica que la realidad en el caso de los estudiantes de la UAIM, no les gusta la lectura, de acuerdo con los datos obtenidos en la entrevista con la maestra, y se constituye como de los factores que impide el aprendizaje de la asignatura.

En la *Figura 5* se puede observar en resumen los temas y categorías relacionados con los problemas en el proceso de aprendizaje de la programación orientada a objetos.

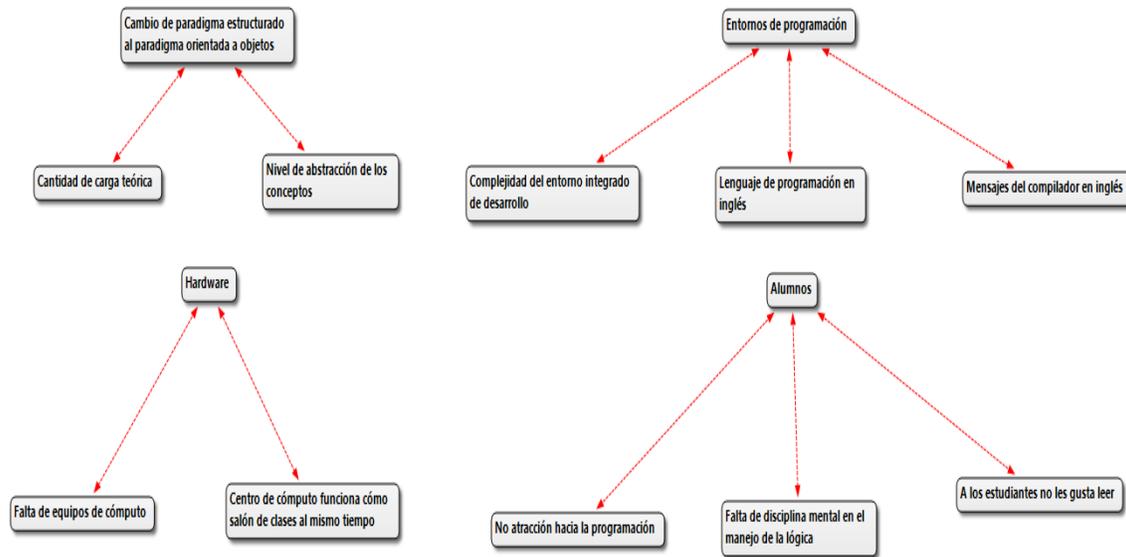


Figura 5.- Temas y categorías relacionados con los problemas en el proceso de aprendizaje de la POO.

CONCLUSIÓN

El objetivo planteado en la presente investigación fue Analizar los problemas en el proceso de aprendizaje de la programación orientada a objetos en los estudiantes de la generación 2012-2016 de la carrera de ingeniería en sistemas computacionales de la UAIM, las principales problemas en el proceso de aprendizaje se han descrito e interpretado en el apartado anterior, de lo cual se desprende que el factor que presentó la mayor frecuencia es la falta disciplina mental en el manejo de la lógica por parte de los estudiantes al momento de programar, esto se debe a que actualmente se les enseña a los estudiantes directamente el lenguaje de programación sin detenerse primero en hacer que los estudiantes reflexionen como resolver el problema de automatización que se les presenta independientemente del lenguaje de programación utilizado, mediante la utilización de metodologías de desarrollo y de modelos utilizando la notación UML (Lenguaje Unificado de Modelado) que les permite resolver los grandes problemas en tiempo de diseño mediante la utilización de la lógica o pensamiento y que contribuya en gran medida a reducir la complejidad inherente en el desarrollo de software tal y como lo señala López (2013)

En la actualidad muchos estudiantes de programación están aprendiendo Java, que es un lenguaje orientado a objetos, sin embargo muchos de ellos, no están

aprendiendo a programar orientado a objetos, porque se les está enseñando prácticamente en forma directa el lenguaje Java y no se les está enseñando a 'pensar', es decir no están desarrollando la lógica de la programación orientada a objetos.

Por lo tanto se puede afirmar con base en la evidencia obtenida que es necesario que los estudiantes cursen una asignatura de metodología de desarrollo previa a la programación orientada a objetos que les permita aprender la lógica implícita en la misma.

LITERATURA CITADA

- Brooks, F. (1987). No Silver Bullet: Essence and Accidents of Software Engineering. IEEE Computer, 20(4), 10-19. Disponible en: <http://www.cs.nott.ac.uk/~cah/G51ISS/Documents/NoSilverBullet.html>
- López, R. (2013). Metodología para el desarrollo de la lógica de la programación orientada a objetos. Sistemas, Cibernética e Informática, 10(2), 27-329. Disponible en: [http://www.iiisci.org/journal/CV\\$/risci/pdfs/CA889XD13.pdf](http://www.iiisci.org/journal/CV$/risci/pdfs/CA889XD13.pdf)
- Ramírez, M., Lucio, M., Garza, J., García, L. y Vargas, J. (2011). "Alice": Un entorno diferente para aprender programación orientada a objetos. CienciaUAT, 22(4), 64-68.
- Rodríguez, D. y Valldeoriola, J. (s.f.). Metodología de la investigación. Universitat Oberta de Catalunya. España. Disponible en: http://zanadoria.com/syllabi/m1019/mat_cast-nodef/PID_00148556-1.pdf
- Spigariol, L. y Passerini, N. (2013). Enseñando a programar en la orientación a objetos. Memoria del 1er. Congreso Nacional de Ingeniería Informática / Sistemas de Información. Educación en Ingeniería. Red de informática / Sistemas de Información del Consejo Federal de Decanos de Ingeniería. Córdoba, Argentina. ISSN: 2347-0372. Disponible en: <http://conaiisi.frc.utn.edu.ar/EI>
- Sousa, D. (2002). Cómo aprende el cerebro. Una guía para el maestro en la clase. Estados Unidos: Corwin Press.
- UAIM. (2011). Informe de la fase de diagnóstico. Programa de formación, actualización y profesionalización docente. Departamento de desarrollo académico. Universidad Autónoma Indígena de México.

AGRADECIMIENTOS

A los titulares académicos y facilitadores educativos de la Universidad Autónoma Indígena de México Unidad Mochicahui y Los Mochis por el tiempo dedicado a esta investigación y por la contestación a las entrevistas realizadas. Así mismo al Programa para el Desarrollo Profesional Docente por el apoyo recibido para la redacción de tesis doctoral toda vez que este artículo es un producto de la misma.

Síntesis curricular

José Emilio Sánchez García

Licenciado en Sistemas Computacionales por la Universidad de Occidente Unidad Los Mochis, Maestro en Sistemas de Información Administrativa por la Universidad de Occidente Unidad Los Mochis en el cual obtuvo distinción por tener el promedio más alto de su generación, C. a Dr. en Tecnología Educativa en el Centro Escolar Mar de Cortés . Actualmente se desempeña como Profesor-Investigador y como Comisionado de Vinculación, Difusión y Extensión de la Investigación en la Universidad Autónoma Indígena de México.

Margarita Urías Ruiz

Ingeniera en Sistemas Computacionales por la Universidad Autónoma Indígena de México Unidad Mochichahui, Maestra en Sistemas de Información Administrativa por la Universidad de Occidente Unidad Guasave. Actualmente se desempeña como Jefe de Carrera del Programa Educativo de Licenciatura en Ingeniería en Sistemas Computacionales Unidad Mochichahui. Co-autora de una publicación en Alemania de un artículo arbitrado en la revista Springer intitulado An Application of a Multicriteria Approach to Compare Economic Sectors: The Case of Sinaloa, México.

Brenda Edith Gutiérrez Herrera

Ingeniera en Sistemas Computacionales por la Universidad Autónoma Indígena de México Unidad Mochichahui, Maestría terminada en Sistemas de Información Administrativa por la Universidad de Occidente Unidad Guasave y en proceso de elaboración de tesis. Actualmente se desempeña como Facilitador Educativo en la Licenciatura de Ingeniería en Sistemas Computacionales Unidad Los Mochis