

Anteproyecto de Trabajo de Grado

Jota Emilio López Ramírez, Esmeralda Rivas Guzmán

Universidad del Valle

Facultad de Ingeniería

jota.lopez@univalle.edu.co, esmeralda.rivas@univalle.edu.co

Palabras clave—Trabajo de grado, Anteproyecto, Investigación, Universidad del Valle, Enseñanza

I. INTRODUCCIÓN

El presente anteproyecto tiene como propósito establecer las bases teóricas y metodológicas del trabajo de grado propuesto. Se expone el contexto general, la motivación y la relevancia del problema a abordar.

El proceso de enseñanza del diseño de compiladores ha sido abordado desde múltiples enfoques pedagógicos. Por ejemplo, Kundra y Sureka [1] presentan un enfoque basado en aprendizaje por proyectos y casos para mejorar la comprensión de los conceptos de compiladores. De manera complementaria, Vegdahl [2] propone el uso de herramientas de visualización para apoyar el aprendizaje de los procesos internos del compilador.

Otros autores como Baldwin [3] destacan la utilidad de lenguajes simplificados y compiladores modulares para facilitar la enseñanza práctica del tema, mientras que Mernik y Žumer [4] desarrollaron una herramienta educativa específica para la construcción de compiladores.

La importancia de los lenguajes de programación como marco conceptual para el aprendizaje ha sido resaltada desde los primeros trabajos de Feurzeig, Papert y Lawler [5], quienes vinculan la programación con la enseñanza de las matemáticas desde un enfoque constructivista.

Finalmente, investigaciones más recientes como la de Stamenković y Jovanović [?] exploran sistemas web interactivos que permiten la enseñanza de compiladores a través de entornos digitales, integrando elementos de visualización, automatización y simulación educativa.

Estas contribuciones conforman el marco de referencia que sustenta la relevancia y viabilidad del presente proyecto, al evidenciar la evolución de las estrategias pedagógicas aplicadas al ámbito de los compiladores y su enseñanza.

II. FORMULACIÓN DEL PROBLEMA

II-A. Descripción del Problema

En la actualidad, las organizaciones enfrentan dificultades relacionadas con XYZ, lo cual impacta negativamente en los procesos de ABC.

II-B. Definición del Problema

A partir del análisis anterior, se formula el siguiente problema de investigación: "¿Cómo puede la implementación de XYZ contribuir a mejorar DEF en el contexto de ABC?"

III. MARCO DE REFERENCIA

III-A. Marco Teórico

Se presentan los fundamentos conceptuales y teóricos relacionados con el área de estudio.

III-B. Estado del Arte

Se analizan las principales investigaciones, proyectos y desarrollos previos que abordan el mismo problema o similares.

III-C. Antecedentes

Se describen experiencias y estudios previos relevantes que sirven como punto de partida para este anteproyecto.

III-D. Marco Conceptual

En este apartado se definen los términos y conceptos clave utilizados a lo largo del proyecto.

IV. ALCANCE DEL PROYECTO

IV-A. Declaración del Alcance

El proyecto se centrará en el diseño, desarrollo y evaluación de XYZ para mejorar los procesos de ABC.

IV-B. Objetivos

IV-B1. Objetivo General: Desarrollar una herramienta interactiva que facilite la comprensión y aplicación de los contenidos de la asignatura *Fundamentos de Interpretación y Compilación de Lenguajes de Programación (LP)* de la Universidad del Valle, mediante la visualización de procesos internos y la generación de lenguajes orientados por sintaxis.

IV-B2. Objetivos Específicos:

- Identificar las dificultades conceptuales y metodológicas que enfrentan los estudiantes de Ingeniería de Sistemas en la asignatura *Fundamentos de Interpretación y Compilación de LP*.
- Caracterizar los elementos teóricos y técnicos necesarios para la construcción de intérpretes orientados por sintaxis que favorezcan su aplicación en contextos pedagógicos.
- Diseñar una estrategia arquitectónica y didáctica para la construcción de intérpretes orientados por sintaxis, integrando representaciones visuales y mecanismos de interacción que promuevan el aprendizaje autónomo.
- Implementar una herramienta interactiva que permita definir gramáticas, visualizar la estructura sintáctica de programas y el seguimiento de su evaluación en los entornos de ejecución.

- Evaluar la funcionalidad técnica y la usabilidad de la herramienta desarrollada mediante pruebas con estudiantes de la asignatura *Fundamentos de Interpretación y Compilación de LP*.

IV-B3. Restricciones y Supuestos: Las siguientes restricciones y supuestos delimitan el alcance y las condiciones del proyecto.

IV-B3a. Restricciones temporales:

- El proyecto debe completarse en el tiempo estipulado para un trabajo de grado, lo que limita el alcance de las funcionalidades y el tiempo para la validación del sistema.
- La evaluación con estudiantes debe realizarse dentro de un período académico específico coordinado con el calendario de la asignatura.

IV-B3b. Restricciones de recursos humanos:

- El desarrollo será realizado por un equipo de dos estudiantes de trabajo de grado, lo cual limita la complejidad y cantidad de características que pueden implementarse.
- La disponibilidad de estudiantes para las pruebas dependerá del ciclo académico y de su participación voluntaria.

IV-B3c. Restricciones tecnológicas:

- La herramienta debe ser accesible sin requerir instalación de software especializado por parte de los usuarios finales, priorizando plataformas web o de fácil acceso.
- El sistema deberá funcionar en equipos con recursos computacionales estándar utilizados por los estudiantes universitarios y el docente.
- No se dispone de infraestructura de servidores institucionales garantizada, por lo que el despliegue se limitará a servicios gratuitos o de bajo costo.

IV-B3d. Restricciones de alcance funcional:

- El sistema se enfocará en la interpretación de lenguajes orientados por sintaxis, sin incluir procesos de compilación avanzada ni optimización de código.
- El sistema soportará un subconjunto representativo de características de lenguajes de programación, sin pretender ser un entorno de desarrollo completo.
- La visualización se centrará en los conceptos fundamentales de la asignatura, sin abarcar todos los temas del curso.

IV-B3e. Restricciones de evaluación:

- No se podrá medir el impacto en el aprendizaje a largo plazo, debido a que ello requeriría estudios longitudinales fuera del alcance temporal del proyecto.
- La evaluación se centrará en aspectos de funcionalidad técnica, usabilidad y percepción de utilidad por parte de los estudiantes y el docente.

IV-B3f. Supuestos del contexto educativo:

- Los estudiantes cuentan con conocimientos básicos en programación y lenguajes de programación.
- Existe interés y disposición por parte de los estudiantes en el uso de herramientas complementarias para el aprendizaje.

- El docente de la asignatura está dispuesto a colaborar en la evaluación de la herramienta y facilitar su uso con estudiantes.

IV-B3g. Supuestos técnicos:

- Las tecnologías seleccionadas permitirán desarrollar visualizaciones dinámicas y comprensibles para los usuarios.

IV-B3h. Supuestos de recursos y disponibilidad:

- Se contará con acceso a los estudiantes y al material didáctico de la asignatura durante al menos un período académico.
- Existirán recursos mínimos de infraestructura para desplegar la herramienta durante la evaluación.

IV-B3i. Supuestos sobre el problema y la evaluación:

- La visualización de procesos internos aporta valor pedagógico y refuerza la comprensión teórica.
- Los criterios de funcionalidad, usabilidad y percepción de utilidad son indicadores válidos para evaluar la calidad del prototipo.

V. CONCLUSIONES

El presente anteproyecto sienta las bases para el desarrollo del trabajo de grado propuesto, orientado a XYZ. Se espera que el resultado contribuya al avance académico y tecnológico dentro del área de ABC.

REFERENCIAS

- [1] D. Kundra and A. Sureka, "An experience report on teaching compiler design concepts using case-based and project-based learning approaches," in *2016 IEEE Eighth International Conference on Technology for Education (T4E)*, 2016, pp. 216–219.
- [2] S. R. Vegdahl, "Using visualization tools to teach compiler design," in *Proceedings of the Fourteenth Annual Consortium on Small Colleges Southeastern Conference*, ser. CCSC '00. Evansville, IN, USA: Consortium for Computing Sciences in Colleges, 2000, p. 72–83.
- [3] D. Baldwin, "A compiler for teaching about compilers," in *Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education*, ser. SIGCSE '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 220–223. [Online]. Available: <https://doi.org/10.1145/611892.611974>
- [4] M. Mernik and V. Zumer, "An educational tool for teaching compiler construction," *IEEE Transactions on Education*, vol. 46, no. 1, pp. 61–68, 2003.
- [5] W. Feurzeig, S. A. Papert, and B. Lawler, "Programming-languages as a conceptual framework for teaching mathematics," *Interactive Learning Environments*, vol. 19, no. 5, pp. 487–501, 2011. [Online]. Available: <https://doi.org/10.1080/10494820903520040>