

## Development of a Tool for Academic Data Processing and Visualization: A Python-Based Proposal

### Desarrollo de una herramienta para el Procesamiento y Visualización de Datos Académicos: Propuesta Basada en Python

Salazar-Uitz, Ricardo Rubén<sup>a</sup>, Ramos-Ramos, Rudy Mauricio<sup>b</sup>, Shih, Meng Yen\*<sup>c</sup> and Lezama-Zarraga, Francisco Román<sup>d</sup>

<sup>a</sup> Universidad Autónoma de Campeche • AGW-7002-2022 • 0000-0003-2307-737X • 416277

<sup>b</sup> Universidad Autónoma de Campeche • NVM5897-2025 • 0009-0007-2506-7104 • 2150106

<sup>c</sup> Universidad Autónoma de Campeche • AAB-4875-2021 • 0000-0001-7475-6458 • 408617

<sup>d</sup> Universidad Autónoma de Campeche • U-1229-2018 • 0000-0003-3397-7881 • 205493

#### SECIHTI classification:

Area: Engineering

Field: Technological Sciences

Discipline: Computer Technology

Subdiscipline: Programming languages

<https://doi.org/10.35429/JOCT.2025.9.22.2.1.8>

#### History of the article:

Received: August 22, 2025

Accepted: December 15, 2025

\* [\[smengyen@uacam.mx\]](mailto:smengyen@uacam.mx)



#### Abstract

This work addresses the challenge of processing student data from multiple .CSV files generated by an Online Preparatory Course. Due to the large number of students and recorded activities, manual processing typically requires several days of work. To solve this, a Python-based tool was developed to efficiently process and visualize the data from these files. The tool is user-friendly and adaptable for academic data analysis, particularly focusing on managing large datasets such as course activity lists. The main outcome of this development is the automation of data loading, processing, and normalization, significantly reducing the time and effort required—from several days to just a few minutes. It provides visual comparisons of student performance, enabling the analysis of diagnostic test results, progress through course activities, and final exam outcomes.

#### Resumen

El presente trabajo da solución a la problemática de procesar los datos en diversos formatos .CSV de un Curso Propedéutico en Línea, con los resultados de los estudiantes, el cual requiere varios días de trabajo manual por la cantidad de alumnos y actividades registradas. Por lo que se desarrolló una herramienta basada en Python para procesar y visualizar datos de estos archivos, fácil de usar y adaptable para el análisis de estos. Se enfoca principalmente en gestionar gran cantidad de datos, como son las listas de actividades académicas que ofrece el curso. El resultado de esta herramienta desarrollada es la automatización de la carga, el procesamiento, la normalización y la reducción drástica del esfuerzo y tiempo invertido de días a unos cuantos minutos, presentando los resultados de los estudiantes permitiendo comparar el examen diagnóstico, el avance en las actividades del curso y el examen final.

Objectives	Methodology	Contribution
Automate the processing of academic data.	Tool developed in Python using pandas, openpyxl, and tkinter.	The developed tool: Saves time by automating data analysis.
Integrate and visualize performance from Khan Academy and Google Forms.	Modular architecture with three controllers.	Reduces errors in data handling. Facilitates the diagnosis of student performance.
Support evidence-based pedagogical decisions.	Processes CSV files and spreadsheets.	Improves teachers' decision-making.

Objetivos	Metodología	Contribución
Automatizar el procesamiento de datos académicos.	Herramienta desarrollada en Python con pandas, openpyxl y tkinter.	La herramienta desarrollada: Ahorra tiempo al automatizar el análisis.
Unificar y visualizar el rendimiento de Khan Academy y Google Forms.	Arquitectura modular con tres controladores.	Reduce errores en el manejo de datos. Facilita el diagnóstico de rendimiento estudiantil.
Apoyar decisiones pedagógicas basadas en evidencia.	Procesa archivos CSV y hojas de cálculo.	Mejora la toma de decisiones docentes.

Academic Data, Python, Processing

Datos Académicos, Python, Procesamiento

**Area:** Development of strategic leading-edge technologies and open innovation for social transformation

**Citation:** Salazar-Uitz, Ricardo Rubén, Ramos-Ramos, Rudy Mauricio, Shih, Meng Yen and Lezama-Zarraga, Francisco Román. [2025]. Frontend and Backend: The new approach to the development of a Web platform for automating the control and administration of degree processes at ITESI. Journal of Computational Technologies. 9[22]1-8: e2922108.



ISSN: 2523-6814 / © 2009 The Author[s]. Published by ECORFAN-Mexico, S.C. for its Holding Taiwan on behalf of Journal of Computational Technologies. This is an open access article under the CC BY-NC-ND license [<http://creativecommons.org/licenses/by-nc-nd/4.0/>]

Peer review under the responsibility of the Scientific Committee MARVID®- in the contribution to the scientific, technological and innovation Peer Review Process through the training of Human Resources for continuity in the Critical Analysis of International Research.



## Introduction

The processing of educational data constantly faces challenges due to the tedious nature of manually handling information from platforms such as *Khan Academy* and assessments conducted through *Google Forms* (Salazar-Uitz *et al.*, 2022).

This task, which must be performed frequently, can take several days or even weeks due to the diversity of formats, lack of structure, and the need to combine metrics from different sources using software such as Excel (Estrada Arjona, 2025). Each time it becomes necessary to analyze students' progress or generate reports for teachers, excessive time must be spent cleaning, organizing, and comparing data. This process is not only inefficient but also delays important pedagogical decisions.

Given this situation, a computational tool in Python was developed to automate this entire process and shorten the analysis time (Guindulain Lebrero, 2015). The proposed solution allows for the immediate processing, unification, and visualization of student performance data from Khan Academy along with results obtained in Google Forms. What previously required days of manual work can now be completed in minutes, without human intervention.

The tool consists of three main modules: one specialized in analyzing data from Khan Academy, another focused on processing results from formal evaluations, and a visualization engine that allows users to compare both data sets clearly and effectively for decision-making. Moreover, it was designed with non-technical educators in mind, making it easy to use and promoting a culture of accessible educational analysis.

This tool arises from a real need to transform a slow and tedious process into one that is agile, automated, and useful. It aims to establish efficiency between digital learning and formal evaluation, while improving how teachers make educational decisions based on data (Estrada Arjona, 2025). According to some studies, having a program that performs these processes automatically can be extremely useful for both teachers and students (Leyva López *et al.*, 2018).

## Methodology

Using academic data from the Khan Academy preparatory course, a new automatic tool was created and tested to assist in data processing and analysis. The goal was to improve the data processing time and visualize the results generated by the application to devote more effort to interpretation.

The test data used correspond to first-year students enrolled in the *Software Engineering* degree for the 2024–2025 academic year who participated in the online preparatory course. This course has been implemented annually at the Faculty of Engineering of the *Universidad Autónoma de Campeche* since 2018 (Canto-Canul *et al.*, 2020). To ensure analysis quality, only students who completed all three types of evaluations were included:

- Activities in Khan Academy (Online Preparatory Course)
- Initial Diagnostic Test (Google Forms)
- Final Exam (Google Forms)

The final sample consisted of 14 students with complete data. The data were obtained directly from the platforms in CSV format, exporting files from Khan Academy containing information for each student. In the current year, 167 activities were considered (Salazar-Uitz *et al.*, 2022). Additionally, spreadsheets automatically generated by Google Forms provided information on diagnostic test results (before the course) and final exam results (after the course).

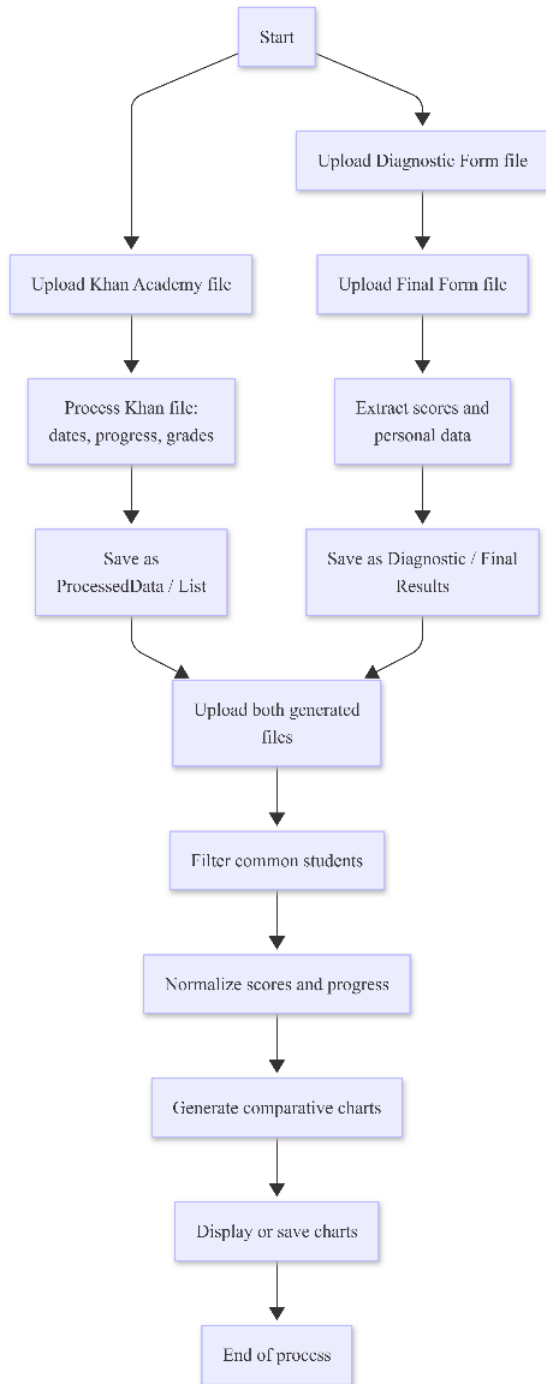
The data processing was carried out on computers running Windows 10 and 11 operating systems. The tool was developed in the Python programming language (version 3.9 or higher), using libraries such as pandas for data manipulation and openpyxl for reading and writing Excel files. The graphical interface was built with tkinter, providing a user-friendly interaction experience.

The tool is composed of three main modules:

- FileController: Processes data exported from Khan Academy.
- GoogleFormController: Handles the analysis of results from Google Forms assessments.

- GraficacionController: Automatically generates comparative visualizations.

**Box 1**



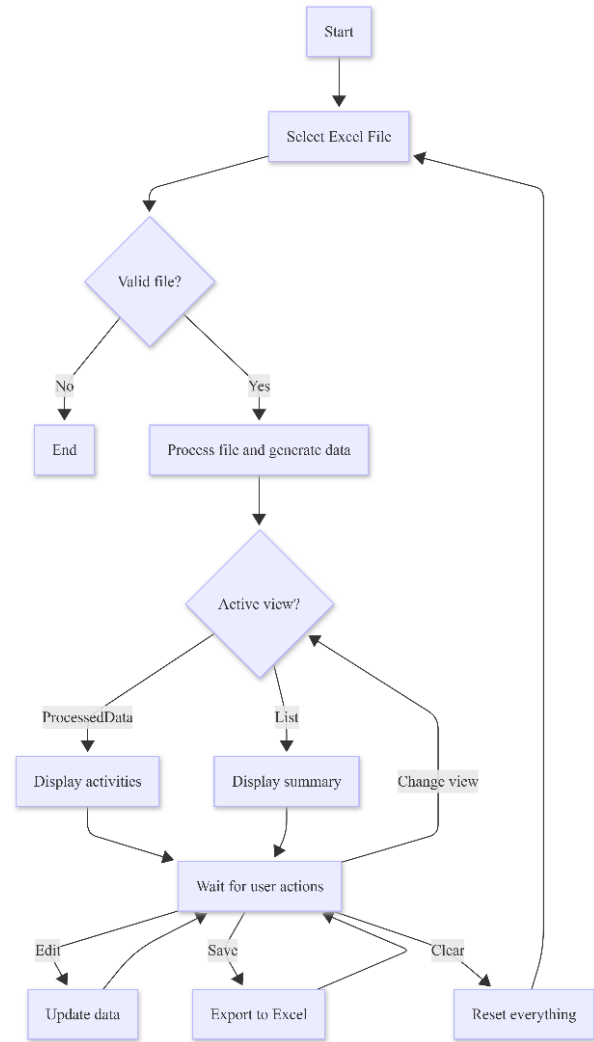
**Figure 1**

General Process Flow Diagram of the Tool for Academic Data Processing

As shown in Figure 1, the tool was designed with a modular approach, allowing the separation of processing responsibilities by platform (Khan Academy and Google Forms), as well as an additional module for graph generation. Each module has an interface aligned with the user’s academic workflow, facilitating file loading, previewing, processing, and exporting of results.

The first component is the FileController, responsible for processing the data exported from the Khan Academy platform. Through a Tkinter-based graphical interface, this module allows the user to select an Excel file and, from it, extract the tasks assigned to students, the completion date for each task, and, optionally, the scores obtained.

**Box 2**



**Figure 2**

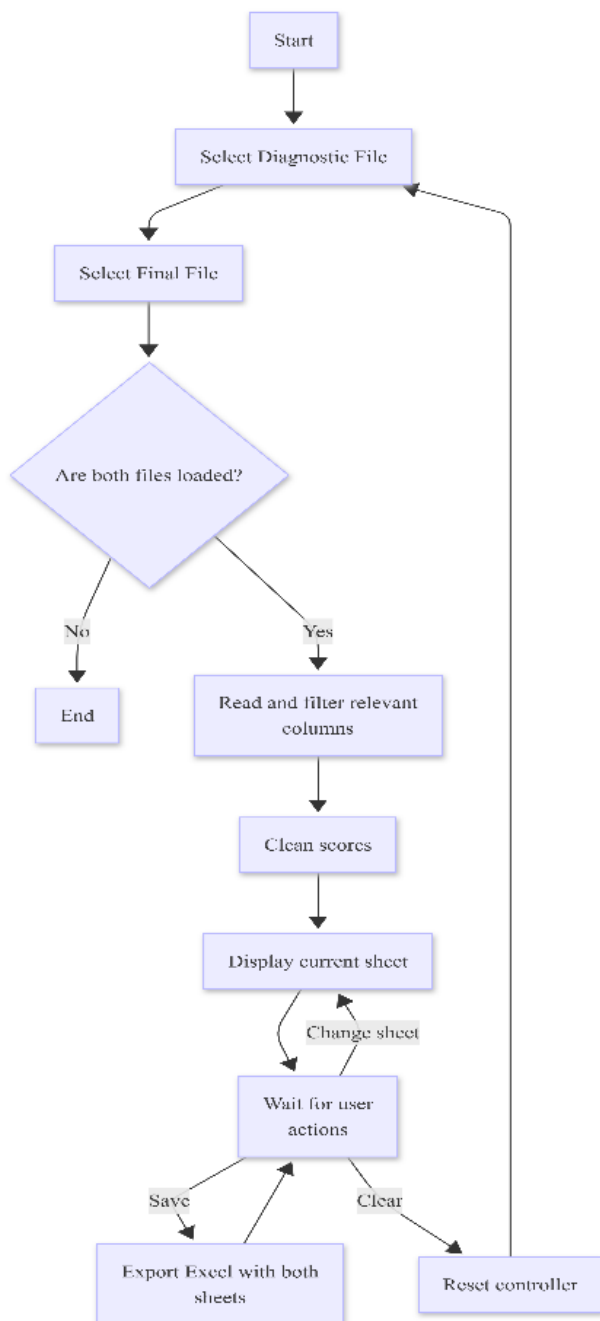
Flow Diagram of the FileController Module: Processing Khan Academy Data

As observed in Figure 2, once the file is loaded a structured table is generated where each row represents an activity and each column corresponds to a student. Cells contain submission dates or are left blank if the activity has not been completed. Subsequently, a second dataset ("Listado") is created, which summarizes each student's performance and includes:

- Number of completed activities.
- Percentage of progress.
- Total grade percentage (if available in the file).

The module also allows switching between these two views and exporting the results to a new Excel file with automatic formulas to compute per-student progress percentages. The second component, named GoogleFormController, is responsible for processing two files coming from Google Forms: one corresponding to an initial diagnostic exam applied at the start of the course and another corresponding to the final exam.

### Box 3



**Figure 3**

Flow Diagram of the GoogleFormController Module: Processing Google Forms Assessment Results.

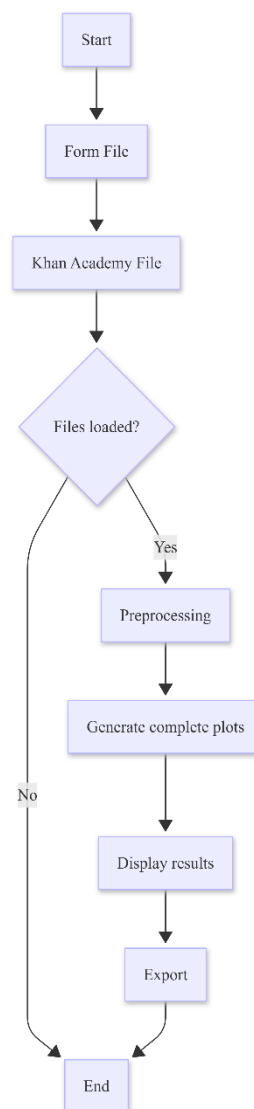
As shown in Figure 3, the program requests both files and extracts the following relevant fields:

ISSN: 2523-6814  
 RENIECYT-SECIHTI: 1702902  
 ECORFAN® All rights reserved

- Username.
- Student's first and last names.
- Score obtained (automatically detected by column name).

During preprocessing, any additional text in score cells is removed, retaining only the numeric value. This allows viewing both diagnostic and final exam results alternately and saving both datasets into a new Excel workbook with separate sheets. The third module, GraficacionController, integrates the datasets processed by the previous two modules to generate graphical representations of student performance. As shown in Figure 4, it requests the files previously produced by the Khan Academy and Google Forms modules. This module identifies common students across both datasets by matching emails or usernames.

### Box 4



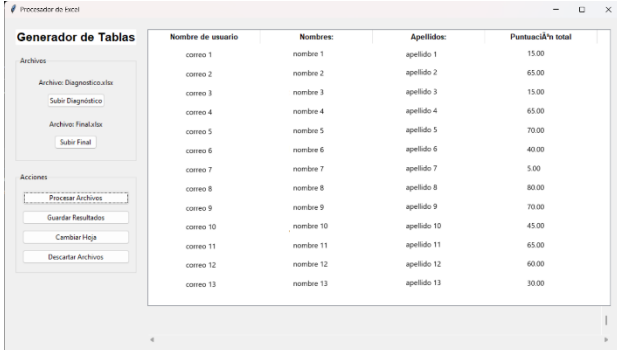
**Figure 4**

Flow Diagram of the GraficacionController Module: Integration and Generation of Comparative Visualizations



Figure 8 illustrates the graphical interface of the GoogleFormController module once the data filtering and analysis process has been completed. This module processes the results of the diagnostic and final exams administered through Google Forms, extracting and organizing the grades for each student into an easily interpretable format.

### Box 8



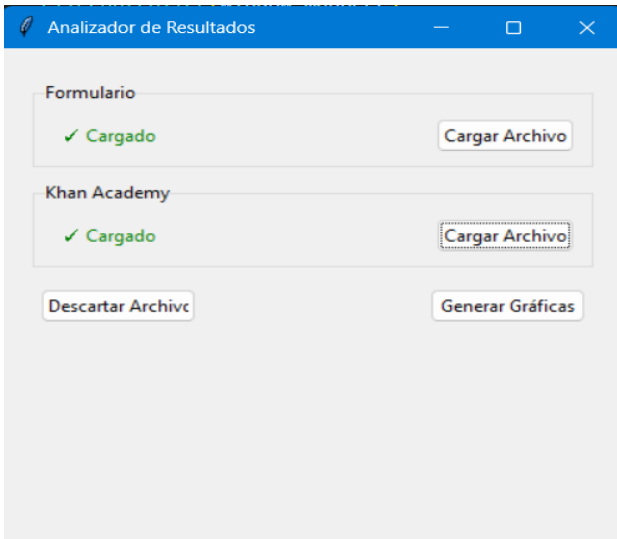
Nombre de usuario	Nombres	Apellidos	Puntuación total
correo 1	nombre 1	apellido 1	15.00
correo 2	nombre 2	apellido 2	65.00
correo 3	nombre 3	apellido 3	15.00
correo 4	nombre 4	apellido 4	65.00
correo 5	nombre 5	apellido 5	70.00
correo 6	nombre 6	apellido 6	40.00
correo 7	nombre 7	apellido 7	5.00
correo 8	nombre 8	apellido 8	80.00
correo 9	nombre 9	apellido 9	70.00
correo 10	nombre 10	apellido 10	45.00
correo 11	nombre 11	apellido 11	65.00
correo 12	nombre 12	apellido 12	60.00
correo 13	nombre 13	apellido 13	30.00

**Figure 8**

GoogleFormController Module Interface Displaying Normalized Results After Assessment Processing.

Figure 9 shows the main interface of the GraficacionController module, where the user must load the previously generated files from the Khan Academy and Google Forms modules. This step is crucial, as the module integrates and compares both datasets to identify common students and generate comparative charts of their academic performance.

### Box 9



Analizador de Resultados

Formulario  
✓ Cargado

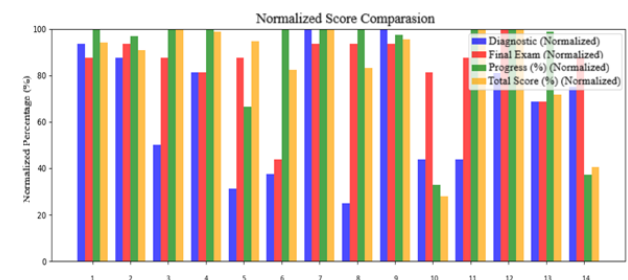
Khan Academy  
✓ Cargado

**Figure 9**

File Upload Interface for the Graphing and Comparative Analysis Module (GraficacionController).

Figure 10 presents a general comparative chart generated by the tool that visualizes student performance across three key stages: the initial diagnostic test, progress and performance in the Khan Academy course, and the final exam. All scores are normalized to a 0–100 range for consistent comparison. This type of visualization facilitates a comprehensive interpretation of academic performance over time, allowing the identification of performance patterns and overall learning evolution.

### Box 10

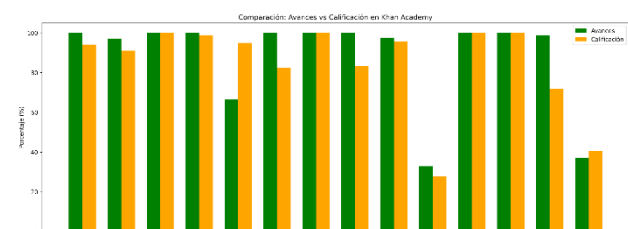


**Figure 10**

General Comparative Chart of Student Performance: Diagnostic, Khan Academy Progress, and Final Exam.

Figure 11 shows a graphical representation comparing the percentage of progress with the total score obtained by students in Khan Academy activities. This graph is critical for analyzing the correlation between task completion and performance on the platform. It helps teachers determine whether progress in assignments directly translates into higher grades, providing valuable insights for adapting pedagogical strategies and enhancing self-paced learning effectiveness.

### Box 11



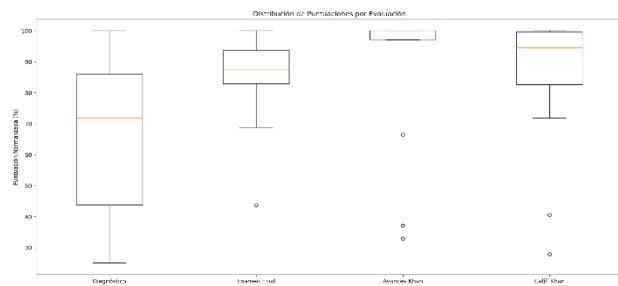
**Figure 11**

Comparative Graph Showing the Relationship Between Progress and Total Score in Khan Academy Activities.

Figure 12 presents box-and-whisker plots (boxplots) representing the distribution of student scores from various data sources, including the diagnostic test, Khan Academy, and the final exam.

This type of chart helps visualize dispersion, median, and quartiles of scores, as well as identify potential outliers. It enables a quick and comparative statistical analysis of group performance, offering deeper insight into the variability and central tendency of academic results.

### Box 12

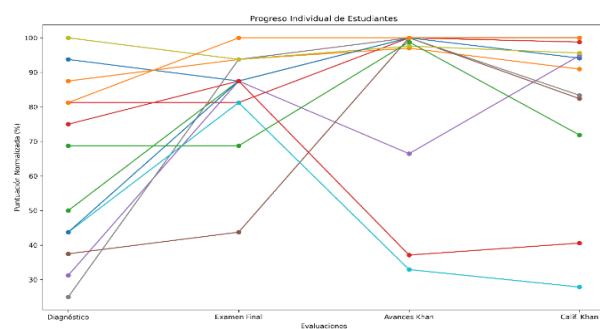


**Figure 12**

Boxplots Showing Score Distribution by Data Source.

Figure 13 displays individualized academic performance graphs per student, showing the learning progression line across the different course evaluations. This visualization is essential for a detailed performance analysis, allowing teachers to track each student's learning trajectory, identify strengths, weaknesses, and specific areas for improvement. It facilitates the adaptation of teaching methods more effectively and is based on evidence supporting each student's needs.

### Box 13



**Figure 13**

Individualized Student Performance Progress Chart.

## Conclusions

This work emerged as a response to the need for integrating and analyzing academic data quickly and automatically from different platforms. Previously, this process required considerable time and effort, as it had to be done manually with spreadsheets, copies, filters, and separate graphs.

To overcome this limitation, a tool was developed in Python using libraries such as pandas and matplotlib to automate the entire process and save time.

With this tool, the time required for data processing can be drastically reduced—from days to just minutes. This not only facilitates academic work but also reduces errors and provides a clearer view of students' progress.

One of the most useful contributions was the ability to generate clear comparative graphs that allow the identification of performance patterns both individually and collectively. The tool was also designed to be easy to use, even for teachers with no programming or data analysis experience. Its graphical interface is simple, requiring no coding skills, making it accessible to a wider audience.

Among the system's main strengths are: complete process automation, direct integration of multiple data sources, clarity in the generated charts, and modular design. However, certain limitations were also identified. For example, the system depends on data being provided in a specific format; if platforms change the export structure, the tool may require adjustments.

In summary, this tool represents a significant advancement in the use of technology to support educational analysis. It has transformed slow, repetitive tasks into fast, automated processes, making evaluation clearer, data-driven, and more useful for improving teaching practices.

## Declarations

### Conflict of interest

The authors declare that they have no conflict of interest.

### Author contribution

*Salazar-Uitz, Ricardo Rubén:* Contributed to the project idea, provided data, and wrote the article.

*Ramos-Ramos, Rudy Mauricio:* Contributed to tool development, data management, and article writing.

*Shih-Meng Yen:* Contributed to formal analysis, writing, and review.

Lezama-Zarraga, Francisco Román:  
Contributed to formal analysis and review.

### Acknowledgements

The Autonomous University of Campeche and the Faculty of Engineering are thanked for the facilities and support granted to develop and implement this research project.

### References

#### Antecedents

Canto-Canul, R. C., López-Martínez, J. L., Salazar-Uitz, R. R., & Lezama-Zarraga, F. R. (2020). *Análisis en retrospectiva del uso de plataformas de aprendizaje virtual como estrategia para evitar la deserción de los estudiantes de nuevo ingreso en Facultades de Ingeniería*. *Revista de Tecnología y Educación*, 4(11), 26–33.

Leyva López, H. P., Pérez Vera, M. G., & Pérez Vera, S. M. (2018). *Google Forms en la evaluación diagnóstica como apoyo en las actividades docentes. Caso con estudiantes de la Licenciatura en Turismo*. *RIDE Revista Iberoamericana Para La Investigación y El Desarrollo Educativo*, 9(17), 84–111.

Salazar-Uitz, R. R., Canto-Canul, R. C., Lezama-Zarraga, F. R., & Shih, M. Y. (2022). *Comparative analysis of the Khan Academy virtual college course to improve new students' academic performance in Faculty of Engineering*. *Journal Practical Didactics*. 2022 12 Article Journal Practical Didactics, 6(11), 11–17.

#### Basics

Estrada Arjona, A. (2025). *Desarrollo de software para el análisis y procesamiento de datos con Python y Excel* [Universitat Politècnica de Catalunya - BarcelonaTech (UPC)].

Guindulain Lebrero, A. (2015). *Software para el procesamiento estadístico de los resultados académicos en la ETSEIB* [ Universitat Politècnica de Catalunya - BarcelonaTech (UPC)].