

EDU METRICS AUTOMATION SYSTEM: AUTOMATED ANALYSIS OF STUDENT AND STAFF PERFORMANCE USING PYTHON

[1] Dr R Kavitha [2] Varshini V [3] Savidha S

[1] Professor, Department of Computer Science and Engineering, Parisutham

Institute of Technology and Science,
Thanjavur, Tamil Nadu – 613006, India

pits.hod.cse@gmail.com

[2] UG Student , Information Technology, Parisutham Institute of Technology and Science,

Thanjavur, TamilNadu-613006, India

vvarshini0706@gmail.com

[3] UG Student , Information Technology, Parisutham Institute of Technology and Science,

Thanjavur, TamilNadu-613006, India

savithasathiya59@gmail.com

Educational institutions require efficient systems to manage and analyze large volumes of academic data. Traditional methods for calculating SGPA, CGPA, and tracking arrears are often manual and prone to errors. This paper presents an Edu Metrics Automation System that processes multi-semester student data using Python and structured Excel inputs. The system automates SGPA and CGPA calculation based on grade points and credits, while also detecting arrears and tracking their clearance across semesters. A key feature is the automatic update of academic performance when arrears are cleared. Additionally, the system evaluates staff performance by computing subject-wise pass percentages. Furthermore, the system generates outputs in both HTML and Excel formats, providing flexibility for visualization and documentation. The Excel output includes multiple sheets for SGPA, CGPA, arrear lists, staff performance, and ranking, enabling easy interpretation and reporting. It reduces manual effort, improves accuracy, and provides comprehensive insights into student and faculty performance. The proposed system is scalable and can be extended for advanced academic analytics using data analysis techniques supported by Python and Pandas libraries [1], [2].

KEYWORDS

SGPA, CGPA, Arrear Detection, Automation, Data Analysis, Academic System.

1. INTRODUCTION

The Edu Metrics Automation System is designed to simplify and automate the process of managing student academic records. In traditional systems, the calculation of SGPA, CGPA, arrears, and performance analysis is performed manually using spreadsheets, which is time-consuming and prone to human errors. This project introduces an automated solution using Python that processes multi-semester data efficiently and generates accurate results. The system enhances productivity by reducing manual work and ensures consistency in academic evaluation.

The system follows a modular architecture consisting of multiple interconnected components. The data upload module accepts Excel files containing semester-wise student data. The processing module extracts subject credits and grades, converting them into grade points. The analysis module performs computations such as SGPA, CGPA, arrear detection, and ranking. Finally, the report generation module produces structured outputs in both HTML and Excel formats. This modular design improves maintainability and scalability of the system. The input to the system is provided in the form of Excel sheets where each sheet represents a semester. Each sheet contains columns such as register number, student name, and subject codes. The first row contains subject credits, and the following rows contain student grades. This structured format allows the system to easily extract and process data.

2. PROBLEM STATEMENT

In many educational institutions, the process of managing and analyzing student academic performance is still carried out using manual or semi-automated methods. Important calculations such as Semester Grade Point Average (SGPA), Cumulative Grade Point Average (CGPA), and arrear tracking are typically performed using spreadsheets or basic tools. This approach requires significant manual effort and becomes highly inefficient when dealing with large volumes of multi-semester student data. Moreover, manual calculations are prone to human errors, which can lead to incorrect results and affect the accuracy of academic evaluation.

Another major issue in the existing system is the lack of an automated mechanism to track arrears and their clearance. When a student fails a subject in one semester and clears it in a later semester, the update process is usually done manually, which is time-consuming and error-prone. There is also no proper system to maintain arrear history or to identify students who have consistently maintained a record of no arrears across all semesters. This limits the ability of institutions to effectively monitor student progress and performance.

In addition, the existing system does not provide integrated analysis of staff performance based on student results. Evaluating pass percentages and teaching effectiveness requires separate manual analysis, which further increases workload. The absence of a unified system that combines student performance, arrear tracking, ranking, and staff evaluation highlights the need for a comprehensive automated solution. Therefore, this project aims to develop an Academic Performance Automation System that can efficiently process multi-semester data, perform accurate calculations, track arrears dynamically, and generate meaningful reports for both students and faculty.

3. PROPOSED METHODOLOGY

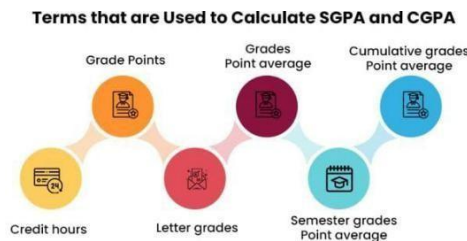
The proposed methodology for the Edu Metrics Automation System is designed to provide a structured and automated approach to processing and analyzing academic data. The process begins with data collection, where student result data is provided in the form of

structured Excel files. Each sheet in the Excel file represents a semester and contains details such as register number, student name, subject codes, credits, and grades. The system is capable of automatically detecting and processing multiple sheets, enabling efficient handling of multi-semester data.

In the data preprocessing stage, the system reads the input data and performs necessary cleaning operations such as handling missing or null values and standardizing column formats. The subject credits are extracted from the first row of each sheet, and student grades are mapped to corresponding grade points using a predefined grading scheme. This step ensures

that the data is in a consistent and usable format for further analysis.

The next stage involves the computation of academic metrics. The system calculates SGPA for each semester by considering the weighted average of grade points and credits. CGPA is calculated as the average of SGPA values across all semesters. During this process, the system identifies failing grades and marks them as arrears. These arrears are recorded and tracked for each student.



A key feature of the methodology is the implementation of automatic arrear clearance detection. The system continuously monitors whether a student who previously failed a subject has passed it in a later semester. When such a case is detected, the system updates the arrear status and recalculates SGPA and CGPA accordingly. This dynamic updating ensures that the academic records always reflect the most recent performance.

The methodology also includes a performance analysis component that evaluates both student and staff performance. Student performance is analyzed based on SGPA, CGPA, ranking, arrear count, and arrear history. The system identifies students with no arrears as well as

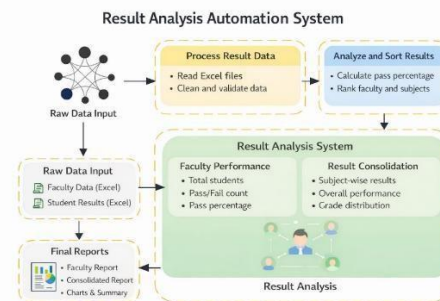
those with consistent performance across all semesters. Staff performance is evaluated by calculating subject-wise and semester-wise pass percentages, which are then mapped with faculty details to assess teaching effectiveness.

Finally, the system generates outputs in both HTML and Excel formats. The Excel output contains multiple sheets for SGPA, CGPA, arrear lists, staff performance, and ranking,

while the HTML output provides a user-friendly interface for quick visualization. This methodology ensures automation, accuracy, scalability, and ease of use, making the system suitable for real-world academic environments.

4. WORKING PRINCIPLE

The system works by reading the input Excel data and mapping grades to corresponding grade points. Based on these values, it calculates SGPA and CGPA using standard formulas. It identifies arrears by checking for failing grades such as RA, U, or F. The system also supports handling missing or null values, ensuring robustness during execution. The system processes each semester sequentially and maintains cumulative data for overall analysis. The workflow ensures that all calculations are automated and consistent across multiple datasets.



5. EXPERIMENTAL SETUP

i) SGPA and CGPA Calculation

The SGPA is calculated by multiplying grade points with subject credits and dividing by the total credits of the semester. The CGPA is computed as the average of SGPA values across all semesters. These calculations are performed automatically by the system, eliminating manual errors and ensuring accuracy. The system dynamically updates CGPA whenever changes occur due to arrear clearance.

ii) Arrear Detection and Clearance

Arrear detection is a key functionality of the system. If a student receives a failing grade, the subject is marked as an arrear. The system tracks these arrears across semesters and identifies whether they are cleared in subsequent attempts. When a student passes a previously failed subject, the system automatically updates the arrear status and recalculates SGPA and CGPA. This feature ensures that the academic record reflects the latest performance.

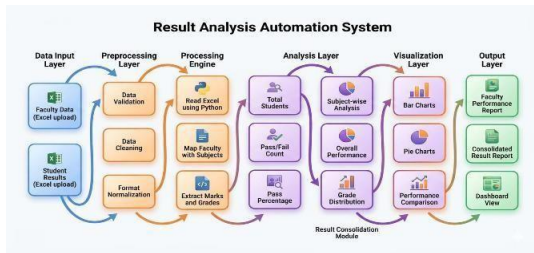
iii) Ranking System

The system calculates both semester-wise and overall ranking based on SGPA and CGPA respectively. Students with equal marks are

assigned the same rank, and subsequent ranks are adjusted accordingly. This ranking system helps institutions compare student performance effectively and identify top performers.

iv) Staff Performance Analysis

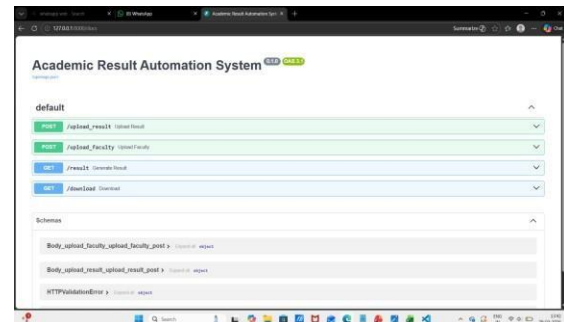
The system includes a staff performance analysis module that calculates pass percentages for each subject and semester.



It determines the number of students who passed and failed and computes the percentage accordingly. This information is mapped with faculty details, enabling institutions to evaluate teaching effectiveness and identify areas for improvement.

6. RESULT AND DISCUSSION

The system successfully processed the input data and generated accurate outputs including semester-wise SGPA, overall CGPA, arrear lists, nil arrear students, and staff performance analysis. The SGPA and CGPA values were computed correctly based on grade points and subject credits, demonstrating the reliability of the calculation module. The system also handled missing and null values effectively without causing execution errors, ensuring robustness. The arrear detection mechanism accurately identified failed subjects for each student and maintained a semester-wise and overall arrear count. A significant outcome of the system is the successful implementation of automatic arrear clearance detection. When a student cleared a previously failed subject in a later semester, the system updated the arrear status and recalculated the SGPA and CGPA dynamically. This ensured that the results always reflected the most recent academic performance.



The ranking module generated both semester-wise and overall ranks based on SGPA and CGPA. Students with equal scores were assigned the same rank, and subsequent ranks were adjusted correctly, ensuring fairness in evaluation.

The system also identified nil arrear students and maintained an arrear history, which helped in recognizing consistently performing students.

In addition to student performance, the staff performance analysis module calculated subject-wise and semester-wise pass percentages. It provided details such as total students, number of students passed, and number of students failed, enabling effective evaluation of teaching performance. The final outputs were generated in both HTML and Excel formats, with Excel containing separate sheets for different analyses. Overall, the system produced accurate, consistent, and well-structured results, demonstrating its effectiveness in automating academic performance analysis. It making the system more scalable and suitable for long-term use.

Student ID	Name	Arrear Status
3.833231e+10	SAVIDHA.S	No Arrear in Entire History
3.833231e+10	SEENIVASAN.S	No Arrear in Entire History
3.833231e+10	SHALINI.P	No Arrear in Entire History
3.833231e+10	VARSHINI.V	No Arrear in Entire History

Arrear Clearance

Staff Percentage

Semester	Subject	Faculty ID	Faculty Name	Total	Pass	Fail	Pass %
sem2	BE2302	F09	Kiruthika	35	28	7	80.00
sem2	CS2321	F08	Poovizhi	35	33	2	94.29
sem2	CS2322	F08	Poovizhi	35	31	4	88.57
sem2	GE2303	F10	Vignesh	35	35	0	100.00
sem2	GE2304	F18	govedan	35	35	0	100.00
sem2	HS2302	F16	Anbuselvi	35	35	0	100.00
sem2	HS2304	F16	Anbuselvi	35	35	0	100.00
sem2	HS2306	F17	tamil	35	32	3	91.43
sem2	MA2302	F12	Bhuvaneshwari	35	34	1	97.14
sem2	PH2304	F11	Vignesh(phy)	35	33	2	94.29
sem5	CS2351	F01	Vinodharasi	34	31	3	91.18
sem5	CS2353	F02	Rajinikanth	34	25	9	73.53
sem5	CS2354	F03	Arunadevi	34	25	9	73.53
sem5	CSV105	F04	Ramya	34	33	1	97.06
sem5	CSV706	F05	Nandhini	34	34	0	100.00
sem5	IT2351	F06	Premalatha	34	34	0	100.00
sem5	MC2301	F07	Clemensiya	34	34	0	100.00

[Download Excel Report](#)

interface and database for better accessibility and data management. Additional improvements such as graphical dashboards, performance visualization, and machine learning-based prediction can further extend its capabilities, making it a comprehensive academic analytics platform.

REFERENCES

[1] G. Van Rossum and F. L. Drake, Python 3 Reference Manual. Scotts Valley, CA, USA: CreateSpace, 2009.

[2] W. McKinney, “Data structures for statistical computing in Python,” in Proc. 9th Python in Science Conf., 2010, pp. 51–56.

[3] I. Sommerville, Software Engineering, 10th ed. Boston, MA, USA: Pearson, 2015.

[4] R. S. Pressman and B. R. Maxim, Software Engineering: A Practitioner’s Approach, 8th ed. New York, NY, USA: McGraw-Hill, 2015.

7. CONCLUSION AND FUTURE ENHANCEMENTS

The Edu Metrics Automation System successfully automates the calculation of SGPA, CGPA, arrear detection, and staff performance analysis, ensuring accuracy, efficiency, and reduced manual effort. The system dynamically updates academic records through automatic arrear clearance detection and provides structured outputs in both HTML and Excel formats, making it practical for real-world academic use. In the future, the system can be enhanced by integrating a web-based

[5] IEEE Standard for Software and System Test Documentation, IEEE Std 829-2008, 2008.

[6] Pandas Documentation. [Online]. Available: <https://pandas.pydata.org>

[7] FastAPI Documentation. [Online]. Available: <https://fastapi.tiangolo.com>

[8] M. Fowler, UML Distilled, 3rd ed. Boston, MA, USA: Addison-Wesley, 2004.

[9] (2024) NumPy Documentation. [Online]. Available: <https://numpy.org>

