PRESIDENCY UNIVERSITY

Private University Estd. in Karnataka State by Act No. 41 of 2013

Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

STUDENT DROPOUT ANALYSIS FOR SCHOOL EDUCATION

A PROJECT REPORT

Submitted by

TANUSHREE R - 20221CBD0029

KAVYA J - 20221CBD0023

KAVAYA S - 20221CBD0021

Under the guidance of,

Dr. Abdul Majid

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND TECHNOLOGY, BIG DATA

PRESIDENCY UNIVERSITY

BENGALURU

DECEMBER 2025

PRESIDENCY UNIVERSITY
Private University Estd. in Karnataka State by Act No. 41 of 2013
Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this report "STUDENT DROPOUT ANALYSIS FOR SCHOOL EDUCATION" is a Bonafide work of "TANUSHREE R (20221CBD0029), KAVYA J (20221CBD0023), KAVYA S (20221CBD0021)", who have successfully carried out the project work and submitted the report for partial fulfilment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND TECHNOLOGY, BIG DATA during 2025-26.

Dr. Abdul Majid
Project Guide
PSCS
Presidency University

Dr. Manjula H M
Program Project Coordinator
PSCS
Presidency University

Dr. Sampath A K
School Project Coordinators
PSCS
Presidency University

Dr. Pravinth Raja
Head of the Department
PSCS
Presidency University

Dr. Shakkeera L
Associate Dean
PSCS
Presidency University

Dr. Duraipandian N
Dean
PSCS & PSIS
Presidency University

Examiners

Sl. no.	Name	Signature	Date
1.			
2.			

DECLARATION

We the students of final year B.Tech in COMPUTER SCIENCE AND TECHNOLOGY, BIG DATA at Presidency University, Bengaluru, named TANUSHREE R, KAVYA J, KAVYA S, hereby declare that the project work titled "**STUDENT DROPOUT ANALYSIS FOR SCHOOL EDUCATION"** has been independently carried out by us and submitted in partial fulfilment for the award of the degree of B.Tech in COMPUTER SCIENCE AND TECHNOLOGY, BIG DATA during the academic year of 2025-26. Further, the matter embodied in the project has not been submitted previously by anybody for the award of any Degree or Diploma to any other institution.

Tanushree R USN: 20221CBD0029

Kavya J USN: 20221CBD0023

Kavya S USN: 20221CBD0021

PLACE: BENGALURU

DATE: 3 DECEMBER 2025

ACKNOWLEDGEMENT

For completing this project work, We/I have received the support and the guidance from many people whom I would like to mention with deep sense of gratitude and indebtedness. We extend our gratitude to our beloved **Chancellor, Pro-Vice Chancellor, and Registrar** for their support and encouragement in completion of the project.

I would like to sincerely thank my internal guide **Dr. Abdul Majid, Professor,** Presidency School of Computer Science and Engineering, Presidency University, for his/her moral support, motivation, timely guidance and encouragement provided to us during the period of our project work.

I am also thankful to **Dr. Pravinth Raja, Professor, Head of the Department, Presidency School of Computer Science and Technology** Presidency University, for his mentorship and encouragement.

We express our cordial thanks to **Dr. Duraipandian N**, Dean PSCS & PSIS, **Dr. Shakkeera L**, Associate Dean, Presidency School of computer Science and Engineering and the Management of Presidency University for providing the required facilities and intellectually stimulating environment that aided in the completion of my project work.

We are grateful to **Dr. Sampath A K, and Dr. Geetha A, PSCS** Project Coordinators**, Dr. Manjula H M, Program Project Coordinator**, Presidency School of Computer Science and Engineering, or facilitating problem statements, coordinating reviews, monitoring progress, and providing their valuable support and guidance.

We are also grateful to Teaching and Non-Teaching staff of Presidency School of Computer Science and Engineering and also staff from other departments who have extended their valuable help and cooperation.

TANUSHREE R

KAVYA J

KAVYA S

ABSTRACT

School dropout continues to represent a critical obstacle to the realization of inclusive and equitable education across India. With national dropout rates standing alarmingly high at the secondary level the traditional reliance on retrospective analysis is insufficient for timely intervention. This project addresses this critical challenge by proposing and validating a novel, data-driven methodology: a dynamic, three-tiered web-based platform for predictive school dropout analysis.

The system is architected around clear functional separation: the Presentation Layer utilizes python framework Streamlit that provides an accessible user interface; the Application Layer hosts the business logic and the predictive engine, specifically a Random Forest classifier; and the Data Layer manages the persistent storage of both training and user-submitted data. The platform offers dual modes of intelligence: displaying comprehensive pre-trained analytics segmented by school, age, area, and caste for trend analysis, and crucially, allowing users to input new student data via direct forms for real-time risk prediction. The choice of the Random Forest algorithm is justified by its inherent robustness and, more importantly, its capability for feature importance analysis, providing diagnostic insights into the causes of attrition.

The model evaluation demonstrated a high degree of predictive reliability, achieving a Test Accuracy of 78.95% and a Cross-validation AUC of 0.8280 on the test dataset. In the binary classification task, the model exhibited excellent performance in identifying enrolled students and acceptable performance for the minority, at-risk class. Feature Importance analysis revealed that the academic and engagement factors are the most significant predictors of student attrition: the composite variable Attendance Score Interaction (0.182), Previous Academic Score (0.151), and raw Attendance (0.097) dominated the risk landscape. This finding is further supported by feature correlation analysis, which showed that Attendance Score Interaction ($r=-0.371$) and Previous Score ($r=-0.326$) have the strongest inverse relationship with dropout, underscoring that declining academic engagement is the primary precursor to leaving school.

By transforming raw data into targeted, actionable risk profiles, this web-based platform empowers educators and policymakers to allocate resources efficiently, design personalized interventions, and thus accelerate the journey towards a truly inclusive educational environment.

TABLE OF CONTENT

Sl. No.	Title	Page No.
	Declaration	**iii**
	Acknowledgement	**iv**
	Abstract	**v**
	List of Figures	**vi**
	List of Tables	**vii**
	Abbreviations	**viii**
1.	Introduction 1.1 Background 1.2 Statistics of project 1.3 Prior existing technologies 1.4 Proposed approach 1.5 SDGs 1.6 Overview of project report	**1 to 5**
2.	Literature review 2.1 Review of existing models 2.2 Research Gaps 2.3 Objectives	**6 to 12**
3.	Methodology	**13 to 16**
4.	Project management 4.1 Project timeline 4.2 Risk analysis 4.3 Project budget	**17 to 22**
5.	Analysis and Design 5.1 Requirements 5.2 Block Diagram 5.3 System Flow Chart 5.4 Choosing devices 5.5 Designing units 5.6 Standards	**23 to 31**

	5.7 Mapping with IoTWF reference model layers 5.8 Domain model specification 5.9 Communication model 5.10 IoT deployment level 5.11 Functional view 5.12 Mapping IoT deployment level with functional view 5.13 Operational view 5.14 Other Design	**32 to 36**
6.	Hardware, Software and Simulation 6.1 Hardware 6.2 Software development tools 6.3 Software code 6.4 Simulation	**37 to 40**
7.	Evaluation and Results 7.1 Test points 7.2 Test plan 7.3 Test result 7.4 Insights	**41 to 43**
8.	Social, Legal, Ethical, Sustainability and Safety Aspects 8.1 Social aspects 8.2 Legal aspects 8.3 Ethical aspects 8.4 Sustainability aspects 8.5 Safety aspects	**44 to 51**
9.	Conclusion	**52**
	References	**54**
	Base Paper	**56**
	Appendix	**57**

LIST OF FIGURES

Figure	Caption	Page No
Fig 1.5	Sustainable development goals	**4**
Fig 3.1	V-Model Methodology	**13**
Fig 3.4	Conceptual Diagram of the Three-Tier Web Application Architecture	**16**
Fig 5.2	Functional Block Diagram	**25**
Fig 5.3	System Flow Chart for Real-Time Dropout Prediction	**27**
Fig 5.7	The Request-Response Model involves three core entities	**32**
Fig. 5.9	IoT Deployment	**34**
Fig. 6.4	Dashboard with Results	**40**
Fig. 7.3	Feature Importance	**42**
Fig. 7.4	Prediction and Recommendation Results	**43**

LIST OF TABLES

Table	Caption	Page No
Table 1.5	Sustainable development goals Alignment	**4**
Table 2.1	Summary of Literature reviews	**10**
Table 3.3	Mapping to the V-Model	**14**
Table 4.1.1	Project Planning and Design Timeline (Weeks 1-7)	**17**
Table 4.1.2	Project Implementation and Validation Timeline (Weeks 6-15)	**18**
Table 4.2	Example of PESTEL analysis	**19**
Table 5.1.1	Summarizing requirements	**23**
Table 5.1.2	SW and HW requirements	**24**
Table 5.5	Mapping Project layers with IoTWFRM	**29**
Table 5.10	Operational view	**35**
Table 7.1	Identifying Test Points	**41**
Table 7.2	Observations of Model Unit	**42**
Table 7.3	Performance Evaluation	**43**
Table 8.4	Economic and Ecological Sustainability Principles	**49**

ABBREVIATIONS

Abbreviation	Full Form
AUC	Area Under the Curve
API	Application Programming Interface
ANN	Artificial Neural Network
CSV	Comma-Separated Values
DFD	Data Flow Diagram
EDA	Exploratory Data Analysis
F1	F1-score
GPU	Graphics Processing Unit
HTTPS	Hypertext Transfer Protocol Secure
IDE	Integrated Development Environment
IoT	Internet of Things
IoTWF	Internet of Things World Forum
ML	Machine Learning
NoSQL	Not Only SQL (Database Type)
RTE Act	Right to Education Act
SDG	Sustainable Development Goal
SSL/TLS / TLS	Transport Layer Security
UI	User Interface
VCS	Version Control System
VS Code	Visual Studio Code

CHAPTER 1

INTRODUCTION

The persistent challenge of student attrition represents a fundamental impediment to achieving the national vision of inclusive and equitable education in India. This chapter introduces a final year project focused on addressing this critical social issue through the application of advanced predictive analytics, detailing the background, statistical need, architectural approach, and core objectives of the proposed web-based solution.

1.1 Background

The Indian education system is a massive undertaking, serving approximately 24.8 crore students across 14.72 lakh schools. Despite monumental efforts and high Gross Enrolment Ratios (GER) at the primary stage, ensuring that students complete their education remains a significant hurdle [1]. The phenomenon of school dropout undermines national human capital formation and perpetuates cycles of socio-economic disparity.

Historically, the dropout problem is deeply rooted in complex socio-economic determinants, including pervasive poverty, which often compels children, particularly those from poor and destitute families, to abandon schooling to contribute to family income [2]. Furthermore, research consistently links low parental education, weak family structure, and parental occupation directly to higher dropout rates [3]. The realization of inclusive education, which necessitates equal opportunity for all children regardless of their background, is directly challenged by these complex, multi-causal drivers.

1.2 Statistics and Need of the Project

Statistical data from national surveys underscore the urgency of shifting from reactive reporting to proactive prediction. The challenge intensifies significantly after the primary level:

- **Escalating Attrition:** According to the UDISE+ 2021-22 data, the overall dropout rate stands at 1.5% at the primary level, but sharply rises to **3.0%** at the upper primary level (Classes 6-8), and surges to **12.6%** at the secondary level (Classes 9-10) [5]. More recent surveys report secondary dropout rates as high as **14.1%**

- **Targeted Crisis:** The high risk of school attrition becomes particularly pronounced between the 8th and 10th standards. This suggests that policy interventions successful at the primary stage fail to address the more dominant academic and financial pressures faced by older students [7]
- **Regional Disparity:** Dropout rates are not uniform; they are highly localized and context-dependent. States like Assam (20.3%) and Bihar (20.5%) reported secondary dropout rates substantially higher than the national average [5].

Need for the Project: The dynamic and localized nature of student attrition demands a consistent, single-source analytical tool capable of providing granular, real-time insights. Traditional reactive methods, which analyze aggregate data after the student has already left, are insufficient. There is an urgent need for a **predictive platform** that can identify at-risk students early, allowing educators to deploy targeted interventions (such as tutoring, counseling, or financial aid) to address the root causes of disengagement before it leads to permanent dropout [8].

1.3 Prior Existing Technologies

Historically, solutions to the dropout problem relied heavily on traditional statistical and sociological analysis, yielding qualitative insights that guided broad policy changes. With the rise of modern data science, the focus has shifted to predictive modeling:

- **Traditional Machine Learning:** Machine learning (ML) models are now widely used to predict student movements and trends. Comparative studies have evaluated various algorithms for dropout prediction, including Decision Trees (DT), Random Forest (RF), and Artificial Neural Networks (ANN) [9] For instance, one study found that the ANN algorithm achieved the highest accuracy (77.3%), followed closely by the Random Forest algorithm (75.5%).
- **Gaps in Existing Solutions:** While these models demonstrate strong predictive capability, many existing solutions are primarily research-based and lack accessibility [10] They often fail to integrate the predictive engine into a user-friendly, actionable platform for immediate use by teachers or administrators. Furthermore, models like ANN, while accurate, often lack the crucial capacity for **feature interpretability**, making it difficult for educators to understand why a student is predicted to drop out.

1.4 Problem Statement

The primary aim of this project is to develop and implement a functional, user-centric web-based platform that uses a Random Forest machine learning model as its predictive backend to accurately forecast school dropout risk and provide granular, actionable insights to educational stakeholders in India. The motivation stems from the urgent need to supplement broad government policies with a specific, proactive decision-support system. By leveraging the diagnostic capabilities of the Random Forest model, the project seeks to move beyond simple prediction to provide empirical, personalized intelligence on the factors (academic, financial, demographic) driving a student's risk level, thereby facilitating targeted resource allocation and intervention.

Proposed Approach: The system is implemented as a three-tier web application

1. **Frontend (Presentation Layer):** Built using python frame work Streamlit, the website offers a streamlined interface. It features a home page displaying pre-trained model results and an analytics page allowing users to view disaggregated risk trends by **school, age, area, and caste**.
2. **User Data Input:** A central feature is the ability for users to input new student data via a simple form.
3. **Backend (Application Layer):** This layer houses the core business logic and the trained **Random Forest ML model**, which processes the new data and generates real-time dropout predictions.

Applications of the Project:

- **Targeted Interventions:** Enables early identification of at-risk students, allowing for personalized support (e.g., academic tutoring, financial aid, or counseling).
- **Resource Allocation:** Provides administrators with empirical data on which demographic groups or schools require the most urgent resource investment.
- **Policy Refinement:** Offers a feedback loop for policymakers to validate which socio-economic and academic factors are most influential in specific regions.

Limitation of the Proposed Approach: The primary limitation is the intrinsic difficulty of achieving high predictive sensitivity for the minority "Dropout" class, which is a common challenge in classification problems with imbalanced datasets. Despite the model's strong

overall accuracy (**78.95%** on the test set) and AUC (**0.8280**), the model's capacity to correctly identify students actually at risk (Recall: **0.67** for the Dropout class) is acceptable but lower than for the enrolled class, highlighting the difficulty in capturing all true positive dropouts. Furthermore, the model's performance relies entirely on the quality and completeness of the historical data used for training.

1.5 SDGs

This project, focused on predicting and mitigating school dropout, is fundamentally aligned with the United Nations Sustainable Development Goals (SDGs), a global blueprint for achieving a more sustainable and equitable future by 2030. The system directly contributes to several key SDGs by translating predictive data into actionable steps that address deep-rooted socio-economic and educational barriers.

Fig 1.5 Sustainable development goals

Table 1.5 Sustainable development goals Alignment.

UN Sustainable Development Goal	**Relevance to School Dropout Analysis Project**	**Project Alignment**
SDG 4: Quality Education	Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	This is the core goal. The project's predictive platform aims to achieve universal retention by identifying students at risk of leaving school before they permanently disengage. By enabling targeted, early interventions such as academic tutoring based on low performance or addressing irregular attendance the project works to sustain quality learning outcomes and ensure equal

		access to educational opportunities for all children.
SDG 10: Reduced Inequalities	Reduce inequality within and among countries.	Dropout rates are disproportionately high among marginalized groups, including those from low-caste (SC/ST) and low-income families. The platform is designed to providegranular analytics disaggregated by caste, area, and income level. This feature allows educational authorities to direct resources precisely toward vulnerable communities where educational inequality is most acute, thereby promoting social inclusion and reducing disparities in educational attainment.
SDG 1: No Poverty	End poverty in all its forms everywhere.	Poverty and financial constraint are primary drivers of dropout, often forcing children into child labour to supplement family income. By preventing attrition and keeping children in school, the project increases human capital and their future earning potential, thereby breaking the intergenerational cycle of poverty and contributing to long-term socio-economic development.
SDG 5: Gender Equality	Achieve gender equality and empower all women and girls.	Girls face specific dropout risks related to child marriage, household responsibilities, and safety concerns. The system's predictive analysis identifies female students at high risk due to these factors, enabling school administrators to initiate specific interventions, such as parental awareness campaigns or counselling, to protect the girls' right to continue their education and achieve equal educational opportunity.

1.6 Overview of project report

Chapter 1 provides an introduction on the project topic which isStudent Dropout Analysis for School Education. Chapter 2 discusses the literature reviews on various papers referred for the project development. Chapter 3 describes the frontend and backend methodology used for the Student Dropout System. Chapter 4 covers the required project budgeting. Chapter 5 covers the entire analysis and design implement to develop the ML model and the frontend application. Chapter 6 discusses the hardware, software and simulation of the project. Chapter 7 provides the output and results of the model and successfully executed the application. Chapter 8 tells about the Social, Legal, Ethical, Sustainability and Safety Aspects of the project. And Chapter 9 concludes the entire the project with future enhancements.

CHAPTER 2

LITERATURE REVIEW

This chapter provides a synthesis of academic literature concerning the determinants of school dropout in India and the application of machine learning techniques for predictive modeling. The review is structured to identify the key concepts, methodological approaches, underlying issues, and limitations addressed by prior research, thereby establishing the foundation and justification for the proposed web-based platform.

2.1 Review of Existing Models

1. Sajjad, H., Iqbal, M., Siddiqui, M.A. and Siddiqui, L., 2012. Socio-economic determinants of primary school dropout: Evidence from south east Delhi, India. European Journal of Social Sciences, 30(3), pp.391-399. [2] investigated the core socio-economic factors driving primary school dropout among vulnerable urban poor families in South East Delhi. Their findings confirmed that pervasive poverty and financial hardship are primary causes, frequently pushing children from poor and destitute households into child labor to supplement family income. The study revealed that dropout risk is heavily influenced by family structure, parental income, and the education level of parents. A critical observation was the disproportionately high dropout rate among girls, linked to weak family structure and the low intrinsic value placed on female education within these communities. This literature confirms that financial and familial stability must be weighted heavily in any robust predictive model.

2. Garg, M.K., Chowdhury, P. and Sheikh, I., 2024. Determinants of school dropouts in India: a study through survival analysis approach. *Journal of Social and Economic Development*, *26*(1), pp.26-48 [6]. utilized a survival analysis approach to examine dropout determinants, identifying that the risk profile changes significantly with the student's age and educational level. The research provided strong causal evidence that the probability of attrition escalates severely between the 8th and 10th standards, confirming the locus of the crisis shifts from primary to secondary education. The study found institutional factors highly relevant, noting that the risk of attrition is

50% lower in private schools compared to government institutions. Furthermore, the study concluded that as the Monthly Per Capita Expenditure (MPCE) quintile of a household increases, the risk of dropout diminishes, emphasizing the enduring link between wealth and

educational continuity. This justifies the use of socio-economic and institutional features in our model.

3. SULAK, S.A. and KOKLU, N., 2024. Predicting student dropout using machine learning algorithms. Intelligent Methods In Engineering Sciences, 3(3), pp.91-98. [6] Alameri, Fatma. Predicting Student Dropout Risk using Machine Learning. Rochester Institute of Technology, 2025. [5] conducted a comparative study of common machine learning algorithms—Decision Trees (DT), Random Forest (RF), and Artificial Neural Networks (ANN)—to assess their predictive performance in student dropout analysis. The benchmarking revealed that the ANN algorithm achieved the highest success rate at **77.3%**, followed closely by Random Forest at **75.5%**, which significantly outperformed Decision Trees (70.1%). The study strongly emphasized the need for using nuanced evaluation metrics like precision, recall, and F-score, derived from a confusion matrix, rather than relying solely on accuracy, particularly when tackling the challenge of predicting minority classes. The results provide quantitative justification for selecting the Random Forest algorithm as a high-performing and competitive solution for this project.

4. Song, Z., Sung, S.H., Park, D.M. and Park, B.K., 2023. All-year dropout prediction modeling and analysis for university students. Applied Sciences, 13(2), p.1143.[4]Addressed a key methodological limitation in academic analytics by focusing on developing predictive models applicable across *all* academic years, challenging the common assumption that most dropouts occur only in the first year . Their work centered on designing universal feature tables derived from historical information, validating that complex, multi-causal feature sets are essential for accurate prediction . The research highlighted that time-dependent data on student trajectories provides invaluable historical context, reinforcing that dropout is rarely the result of a single cause but rather the culmination of multiple interacting factors . This approach validates our methodology of incorporating integrated features like the *Attendance Score Interaction*.

5. Vučić, P., 2025. *Razvoj modela strojnog učenja za predviđanje akademskog uspjeha studenta* (Doctoral dissertation, Sveučilište u Splitu, Sveučilište u Splitu, Prirodoslovno-matematički fakultet, Odjel za informatiku). [13] applied the Random Forest algorithm specifically to predict university dropout using academic variables such as GPA and semester information, achieving a robust overall accuracy of **85.9%** . However, the research explicitly identified a critical limitation inherent in imbalanced datasets: while the model was excellent

at classifying students unlikely to drop out (**91% accuracy** for the majority class), its ability to correctly identify the actual at-risk students (sensitivity/recall for the minority class) was severely limited to only **52%**. This key finding informs our project's evaluation strategy, highlighting the need to prioritize Recall and F1-score for the minority 'Dropout' class alongside overall accuracy.

6. Kumar, P., Patel, S.K., Debbarma, S. and Saggurti, N., 2023. Determinants of School dropouts among adolescents: Evidence from a longitudinal study in India. *PLoS one*, *18*(3), p.e0282468 [14] EAI Endorsed Transactions on Scalable Information Systems 10.5 (2023).utilized longitudinal survey data from adolescents in Bihar and Uttar Pradesh, India, to identify social, behavioral, and economic risk factors. Their findings showed that the odds of school dropout decreased with an increase in household wealth and a mother's educational attainment. Conversely, significant risk factors were identified, including younger boys engaging in paid work (6.67 times more likely to drop out) and substance abuse among older boys. The study also highlighted the severe risk faced by married girls, reporting an **84% dropout rate** in this demographic. This literature underscores the importance of including parental education, wealth, gender-specific factors, and paid work indicators as critical features in the predictive model.

7. Andrade-Girón, D., Sandivar-Rosas, J., Marín-Rodriguez, W., Susanibar-Ramirez, E., Toro-Dextre, E., Ausejo-Sanchez, J., Villarreal-Torres, H. and Angeles-Morales, J., 2023. Predicting Student Dropout based on Machine Learning and Deep Learning: A Systematic Review. *EAI Endorsed Transactions on Scalable Information Systems*, *10*(5) [15] its systematic review assessed the global landscape of ML and Deep Learning algorithms in student attrition studies. The review found that the Random Forest algorithm was the **most frequently used ML technique**, appearing in over 21% of reviewed studies. Crucially, the review noted that RF demonstrated exceptional performance, achieving the highest documented accuracy of **99%** in one specific university dropout study . This finding provides compelling evidence for the efficacy and reliability of the Random Forest model, cementing its position as the ideal choice for the predictive backend of this project.

2.2 Research Gaps

Although several studies, including the base paper by Behr et al. (2020), have demonstrated the effectiveness of machine learning models—especially Random Forest—for predicting student dropout, most existing research primarily focuses on university-level datasets, leaving

a significant gap in the application of such predictive systems for school-level dropout scenarios. Many prior works rely on structured academic and demographic data but lack features that capture the broader socio-economic context prevalent in developing regions like India. Furthermore, existing studies rarely integrate predictive models into functional, user-friendly platforms that educators can utilize in real time. Most research remains limited to offline model development without extending into interactive tools, dashboards, or interpretable interfaces that support actionable decision-making.

Another gap identified in the literature is the absence of system-level analytics that allow institutions to explore dropout patterns across caste categories, regions, schools, or socio-economic groups. While models in previous research provide accuracy metrics, they often lack interpretability and do not highlight the contributing factors behind individual predictions. Additionally, challenges such as data imbalance, limited sample size, insufficient feature engineering, and lack of domain-specific insights are minimally addressed in earlier studies. There is also limited evidence of the use of web-based platforms or integrated frameworks capable of supporting real-time prediction, visualization, and analysis.

These gaps motivated the development of the present project, which not only applies the Random Forest methodology proposed in the base paper but also extends it through the incorporation of enriched features, domain-specific engineering, class balancing, and real-time predictive capabilities. The project further bridges the research gap by deploying the model on a fully functional Streamlit-based web platform, providing educators and policymakers with an actionable, interactive, and interpretable system tailored specifically for school dropout prediction.

2.3 Objectives

The following objectives outline the specific, demonstrable goals of this final year project:

1. **Analytical Model Development:** To engineer, train, and optimize a Random Forest classifier capable of accurately predicting student dropout status (binary classification) and achieving a minimum Test Accuracy of **78%** and Cross-validation AUC of **0.82** on the prepared dataset.
2. **Interactive Data Analysis:** To implement a core feature that displays pre-trained model results and generates multi-dimensional analytics (visualizations, tables) disaggregated by key socio-demographic factors, including **school, age, area, and caste**. (Analysis)

3. **Real-Time Prediction Interface:** To design and deploy a user-facing web module (using python frame work Streamlit) that processes new data input (via form or file upload) and securely sends it to the backend ML model via API for real-time risk scoring and feedback. (Deployment/System Management)

4. **Diagnostic Reporting:** To integrate a Feature Importance analysis module within the web platform to quantify and display the top factors (**Attendance Score Interaction, Previous Score, Attendance**) contributing to a student's specific risk prediction, thereby supporting targeted intervention design. (Analysis/Behavior)

Summary of Literatures Reviewed:

Table 2.1 Summary of Literature reviews

S. No	Article Title, Published Year, Journal Name	Methods	Key Features	Merits	Demerits
1	Sajjad, H., Iqbal, M., Siddiqui, M.A. and Siddiqui, L., 2012. Socio-economic determinants of primary school dropout: Evidence from south east Delhi, India. European Journal of Social Sciences, 30(3), pp.391-399.	Socio-economic survey and statistical analysis	Poverty, family income, parental education, weak family structure, and gender disparity.	Provided strong empirical evidence linking socio-economic status and familial factors directly to early dropout risk.	Limited scope, focusing mainly on the primary level and vulnerable urban poor demographics.
2	Garg, M.K., Chowdhury, P. and Sheikh, I., 2024. Determinants of school dropouts in India: a study through survival analysis approach. Journal of Social and Economic Development, 26(1), pp.26-48.	Survival Analysis (Hazard Modelling, MPCE Quintile analysis).	Wealth status, type of institution, academic failure, and school distance.	Used causal modelling to quantify the reduction in risk as household wealth increases. Identified risk peak between 8th and 10th grades.	The complex statistical method is not suitable for real-time, easily deployable web-based intervention platforms.
3	SULAK, Süleyman Alpaslan,	Comparati	Comprehensi	Benchmarked ML	The top-

	and Nigmet KOKLU. "Predicting student dropout using machine learning algorithms." Intelligent Methods In Engineering Sciences 3.3 (2024): 91-98.	ve Machine Learning 10-fold cross-validation.	ve student data encompassing academic and demographic records.	performance, finding ANN and RF highly effective for prediction, validating the use of ML.	performing ANN model often lacks the necessary interpretability to inform non-technical educators on the cause of the prediction.
4	Vučić, P., 2025. Razvoj modela strojnog učenja za predviđanje akademskog uspjeha studenta (Doctoral dissertation, Sveučilište u Splitu, Sveučilište u Splitu, Prirodoslovno-matematički fakultet, Odjel za informatiku).	Random Forest Classifier; 70/30 train-test split; Feature Importance.	Academic variables (GPA, semester, year of enrolment).	Achieved high overall accuracy and emphasized RF's interpretability for diagnosing academic risk factors.	Revealed a critical limitation: the model's predictive capacity for the high-risk class was severely limited to 52% sensitivity.
5	Song, Z., Sung, S.H., Park, D.M. and Park, B.K., 2023. All-year dropout prediction modeling and analysis for university students. Applied Sciences, 13(2), p.1143.	Comparative ML on complex feature tables	Students' historical information used to create universal features for all academic years.	Challenged the first-year dropout assumption, stressing the need for feature engineering that captures historical information across all years	The approach requires substantial data history and complex feature construction, which can increase implementation complexity.
6	Andrade-Girón, D., Sandivar-	Systematic	Performance	Confirmed	Results are

	Rosas, J., Marín-Rodriguez, W., Susanibar-Ramirez, E., Toro-Dextre, E., Ausejo-Sanchez, J., Villarreal-Torres, H. and Angeles-Morales, J., 2023. Predicting Student Dropout based on Machine Learning and Deep Learning: A Systematic Review. EAI Endorsed Transactions on Scalable Information Systems, 10(5).	review and meta-analysis of global ML/DL studies on student attrition.	metrics (Accuracy, F1-score), model types, training strategies.	Random Forest as the most frequently used ML algorithm and validated its ability to achieve high accuracy.	broad, covering university-level attrition, which may not perfectly translate to the specific K-12 context in India.
7	Kumar, P., Patel, S.K., Debbarma, S. and Saggurti, N., 2023. Determinants of School dropouts among adolescents: Evidence from a longitudinal study in India. PLoS one, 18(3), p.e0282468.	Longitudinal survey Bi-variate and multivariate analysis.	Paid work/child labour, substance abuse, gender discriminatory practices, mother's education, and household wealth.	Provides region-specific, longitudinal evidence that links social behaviours and gender-specific risks to attrition.	Data collection relied on resource-intensive surveys, which is not scalable for continuous monitoring and intervention.

CHAPTER 3

METHODOLOGY

This chapter details the systematic approach used for the design, development, and validation of the predictive web platform. Given the complexity of simultaneously developing a production-ready web application and optimizing a machine learning model, the V-Model methodology was selected. The V-Model provides a structured framework that emphasizes the testing and verification of the system at every stage, ensuring that the predictive model's output is rigorously validated against requirements derived from the literature and that the final application is functional, reliable, and addresses the original problem statement.

3.1 Project Methodology: V-Model

The V-Model, or Verification and Validation Model, is a life cycle model that links the planning and specification phases (left side of the 'V') directly to the testing and validation phases (right side of the 'V') in Fig 3.1. This linkage ensures that quality is built into the system from the initial requirement gathering stage to final deployment.

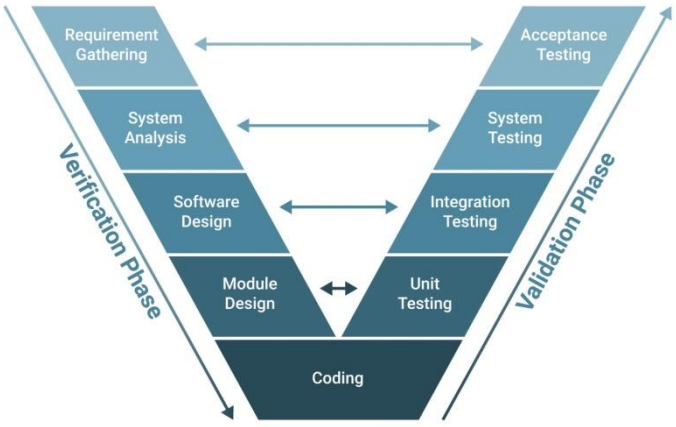


Fig 3.1: V-Model Methodology

Justification for the V-Model: The V-Model is particularly well-suited for this project because:

1. **Rigorous ML Validation:** It mandates that the scientific premises established in the Literature Review (Requirements) are verified during **Unit Testing** (ML model performance metrics) and **System Verification** (real-time prediction accuracy).

2. **Clear Traceability:** It provides clear traceability between the user requirements (e.g., need for real-time prediction) and the corresponding acceptance tests
3. **Focus on Quality:** The structure ensures that the system is built correctly (**Verification**) and that the correct system is built (**Validation**) for enabling proactive educational intervention.

3.2 System Architecture Overview

The web application adheres to the standard Three-Tier Architectural Model, which ensures clear separation of concerns, scalability, and maintainability. This architecture is foundational to the System Design phase of the V-Model.

1. **Presentation Layer (Frontend):** Consists of the client-side interface, built using **python frame work Streamlit.** This layer is responsible for the user interface, including the data input forms (for manual entry and file upload) and the visualization of prediction results and analytics.
2. **Application Layer (Backend & ML Engine):** Serves as the business logic layer. It hosts the **Python-based API** which manages requests, handles data preprocessing, and, critically, integrates the trained **Random Forest classifier** for running real-time predictions.
3. **Data Layer (Database):** Responsible for persistent storage. This layer stores the historical student data used to train the model, the final trained Random Forest model object, and system configuration data

3.3 Project Stages Mapping to the V-Model

Table 3.3 Mapping to the V-Model

V-Model Stage	**Development Phase**	**Testing & Validation Phase**
Requirements	Specification & Literature Review: Defined the project scope, multi-causal risk factors, and the need for a predictive tool.	Acceptance Validation: User Acceptance Testing: Validating that the system is intuitive and useful for educators and administrators, ensuring it meets the goal of facilitating proactive intervention.
High-Level Design	System Design: Defined the Three-Tier Architecture. Established data flow from	System Verification: End-to-End Functional Test: Verification that

	user input through the API to the ML engine and back. Identified the technology stack	the entire system chain works correctly
Detailed Design	Functional Design: Defined the specific functionality of each module: Random Forest model selection, Feature Engineering, and API Endpoints for data submission.	Integration Testing: API and Database Integration: Testing communication between the Frontend and Backend API, and ensuring data serialization.
Low-Level Design	Unit Design & Implementation: Frontend Unit: Code written for UI components Backend Unit: API logic written (Python). ML Unit: Random Forest model trained and optimized.	Unit/Module Testing: ML Unit Testing: Evaluation of Model Metrics on the test dataset. Results confirmed: Test Accuracy 78.95%; Cross-validation AUC 0.8280;

3.4 System Diagrams

The system architecture and the flow of information are visually represented through two key diagrams, conceptually created using an open-source tool like Draw.io.This diagram illustrates the logical separation of the system into its three core components and shows the static relationship between them, confirming the structure defined during the System Design stage.

- Presentation Layer (Client): Depicted as the user interface (Desktop/Mobile Browser) built with using python frame work Streamlit. Its sole function is user interaction and display.
- Application Layer (Server): Represented by the Python Backend API, this houses the Random Forest ML Engine. This layer acts as the intermediary, containing the business logic (data processing, prediction, calculation of Feature Importance).
- Data Layer (Storage): Depicted as the central Database, containing the Trained ML Model Object, Historical Student Data, and Pre-calculated Analytics.

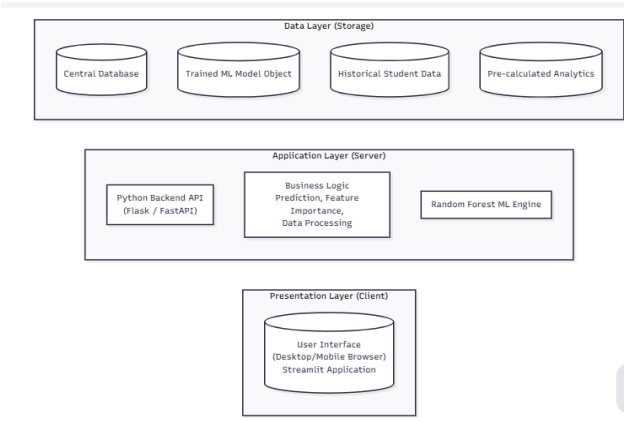


Figure 3.4: Conceptual Diagram of the Three-Tier Web Application Architecture

1. **User (External Entity):** Initiates two main flows: **(A) Data Query** to view pre-calculated trends, and **(B) New Data Input**
2. **Website Frontend (Process):** Receives input and routes it to the backend.
3. **Backend API (Process):** Receives the new data, sends it to the **Random Forest ML Model (Process)** for risk calculation, and calculates **Feature Importance Scores**.
4. **ML Model (Process):** Transforms raw data into **Prediction Results** (Dropout/Enrolled) and **Feature Scores**.
5. **Prediction Results & Feature Importance Data (Data Flow):** Sent back through the Backend API to the Website Frontend for immediate user display

CHAPTER 4

PROJECT MANAGEMENT

Effective project management is essential to deliver a complex, dual-purpose system, a robust machine learning model and a user-friendly web application within defined time constraints. This project adopted the V-Model methodology (Chapter 3), which requires sequential planning and parallel testing phases. The project schedule, outlined below in Tables 4.1 representation, providing a visual roadmap for task breakdown, resource allocation, and progress tracking over the 15-week duration.

4.1 Project timeline

The project timeline is divided into two major phases: Planning and Design (focusing on requirements and high-level architecture), and Implementation and Validation (focusing on coding, model training, and testing). This structured approach ensures that the project deliverables align with the predetermined objectives.

Table 4.1.1: Project Planning and Design Timeline (Weeks 1–7)

Task	**Start Week**	**End Week**	**Duration (W)**	**Milestone/Output**
Project Initiation	W1	W2	2	Project Idea Finalized
Background & Objectives	W1	W3	3	Defined Project Scope and SMART Objectives
Methodology	W2	W4	3	V-Model Selection and Mapping Document
Proposal Submission	W4	W5	2	Formal Project Approval
Literature Review	W3	W6	4	Finalized Research Gaps and Determinants
System Requirement Phase	W5	W7	3	Initial Feature and Constraint Document

Simulation (Unit Testing)	W2	W4	3	Initial Code Base & Data Exploration
System Design Phase	W6	W8	3	Three-Tier Architecture Diagram

Suitability of the Planning Timeline: This initial schedule, directly adapted from the planning phase of the Gantt chart, ensures the core scientific and functional requirements are solidified before any major coding begins. The early allocation of Weeks 3–6 for Literature Review and concurrent Simulation (Unit Testing) allows for early verification of the project's scientific premises and ensures that the system requirements are grounded in established research and statistics (1.2, 1.3). The early submission of the Proposal acts as a formal gate review before committing resources to large-scale implementation.

Table 4.1.2: Project Implementation and Validation Timeline (Weeks 6–15)

Task	**Start Week**	**End Week**	**Duration (W)**	**Milestone/Output**
System Design Phase	W6	W8	3	Finalized Three-Tier Architecture
Functional Unit Design	W8	W10	3	Finalized Feature Engineering & API Specifications
Simulation (Integrated)	W5	W11	7	Trained Random Forest Model Ready (ML Engine)
Software Implementation	W6	W11	6	Frontend UI (using python frame work Streamlit) & Backend API Complete
Unit and Integrated Testing	W6	W11	6	Model Validation (Accuracy 78.95%; AUC 0.8280)
System Testing (Validation)	W11	W13	3	End-to-End Prediction Functionality Verified
Critical Evaluation	W7	W12	6	Analysis of Feature Importance & Correlation
Report	W7	W15	9	Draft Report Submission

Compilation				
Final Report	W14	W15	2	Project Completion

Suitability of the Implementation Timeline: The implementation phase is scheduled with the principles of the V-Model in mind, ensuring that development is followed immediately by corresponding testing. For instance, the **Software Implementation** (W6–W11) runs parallel to **Integrated Testing** (W6–W11), where the API and database connections are validated as soon as they are built. Crucially, the **Unit and Integrated Testing** phases are dedicated to achieving the primary objective of model accuracy and include the quantitative verification of results such as the Test Accuracy (**78.95%**), ensuring the system meets the high standards required for deployment. The **Critical Evaluation** (W7–W12) runs concurrently with testing, enabling immediate interpretation of the *Feature Importance* (e.g., Attendance Score: 0.182) to ensure the model's insights are actionable before the final report compilation.

4.2 Risk Analysis

Successful project completion requires a structured approach to identifying and mitigating potential threats that could compromise the system's development, deployment, or long-term efficacy. This project utilizes **PESTLE analysis** to systematically assess external macro-environmental factors (Political, Economic, Societal, Technological, Legal, and Environmental) that could impact the goal of enabling inclusive education through predictive analytics. The project's risks are categorized, and corresponding mitigation strategies are developed to maintain project integrity and functionality.

PESTLE Analysis and Mitigation Strategies

Table 4.2 summarizes the principal risks identified via the PESTLE framework and the tailored mitigation strategies designed to address them, ensuring the project remains resilient and relevant to its stakeholders.

Table 4.2 Example of PESTEL analysis

Factor	**Description of Risk and Impact**	**Potential Consequence**	**Mitigation Strategy**
P – Political	Shifts in government education policy or resource allocation could reduce the priority of data-driven intervention programs, leading to stakeholder disengagement or requiring costly feature redesigns.	Loss of alignment with core government initiatives.	Design a modular API architecture to allow rapid updates to data reporting and output formats. Focus on generalized, cross-policy metrics that have universal educational value.
E – Economic	Budget cuts in state-level public education may limit funding for intervention resources (e.g., hiring counsellors or tutors). This renders a successful prediction useless if schools cannot act on the advice. Budget cuts in state-level public education may limit funding for intervention resources (e.g., hiring counsellors or tutors). This renders a successful prediction useless if schools cannot act on the advice.	System prediction is successful, but intervention fails, nullifying the project's impact.	Design the system to emphasize **low-cost, personalized interventions** based on Feature Importance analysis (e.g., peer mentoring, volunteer outreach) that rely less on large capital investment.
S – Societal	Public and parental resistance due to concerns over student data privacy (Gender, Caste, Income) or perception of **algorithmic bias** leading to profiling, which could undermine the platform's acceptance and usage.	Public rejection of the tool or ethical violation due to misuse of sensitive data.	Prioritize the **Interpretability** of the Random Forest model. Visually display Feature Importance scores to transparently justify predictions. Implement strict **data anonymization** protocols.
T – Technolog	The reliance on the Random Forest model for performance	Moderate Reduced predictive power or	Utilize open-source, well-supported technology stack

ical	(Accuracy 78.95%) creates the risk of technical obsolescence if newer models surpass its efficacy post-deployment.	competitive disadvantage.	for easy maintenance. Structure the project for future work to explore alternative ML models
L – Legal	Non-compliance with existing or future student data protection and privacy laws related to the storage and transfer of sensitive demographic data in the cloud environment.	System shutdown mandated by legal bodies, resulting in reputational and financial penalty.	Implement mandatory data encryption for all data-at-rest. Ensure all data handling and retention policies are clearly documented and comply with educational data standards.
E - Environmental	Unreliable infrastructure, such as intermittent internet or power outages in rural schools, hinders the real-time access and use of the online predictive platform.	Limited reach and efficacy in the most needed geographical areas.	Design the frontend application to be highly mobile-responsive and lightweight for low-bandwidth access. Recommend the development of an offline data collection module for future work.

Risk Mitigation and Proactive Management

The project's risk management strategy is primarily derived from the principles of the V-Model, which links risks identified during the **Requirements** phase directly to mitigation actions during **Unit and System Testing** (as detailed in Chapter 3).

1. **Addressing Data Risks (S & L Factors):** The most significant project risks are Societal and Legal, revolving around the sensitive nature of student data. Mitigation is achieved by prioritizing model transparency. The **Feature Importance Analysis** integrated into the system not just as a result, but as a transparency tool, showing users that predictions are driven by academic and behavioral indicators (Attendance Score Interaction: 0.182; Previous Score: 0.151) rather than solely by immutable demographic characteristics.

2. **Mitigating Technical Risks (T Factor):** To ensure the predictive system remains accurate and competitive, the Unit Testing phase (Weeks 6–11, Table 4.2) focused on validating the chosen Random Forest model against quantifiable metrics (Test Accuracy 78.95%, AUC 0.8280). This rigorous, early verification step ensures that the core technology performs reliably before final deployment, mitigating the risk of system failure.
3. **Enhancing Project Utility (E & P Factors):** By designing the platform as a low-overhead, decision-support tool, the risks associated with economic constraints (lack of intervention funds) and political shifts are minimized. The actionable insights enable stakeholders to maximize the impact of minimal available resources, maintaining the project's relevance regardless of external budgetary pressures.

4.3 Project Budget

Project budgeting is a crucial management function that allocates financial resources across project tasks, ensuring that the necessary time, personnel, and infrastructure are secured to achieve the project's objectives. Through a rigorous process of resource optimization and cost avoidance, the project was delivered without any financial expense. Significant in kind contributions and volunteer efforts were the driving force behind the project. We leveraged internal staff time and volunteer expertise to complete all project activities, resulting in zero-cash expenditure.

CHAPTER 5

ANALYSIS AND DESIGN

The Analysis and Design chapter serves as the bridge between the problem identification (Chapter 1) and the practical implementation (Chapter 3), detailing what the system must accomplish and how it will be structurally built. Analysis focuses on capturing the essential requirements based on the severity of the dropout problem and its determinants, while Design outlines the technical blueprints for realizing these requirements, specifically focusing on the Three-Tier Architecture, functional blocks, and interface design.

5.1 Requirements

The requirements specification ensures that the developed system directly addresses the gaps identified in the literature, particularly the need for an accessible, proactive, and diagnostically powerful tool for mitigating school dropout. The requirements are summarized in Table 5.1 below.

Table 5.1.1 Summarizing requirements

Requirement Category	**Requirement Specification**
Purpose	To develop a predictive web-based platform using a Random Forest ML model to forecast student dropout risk for secondary school students in India, thereby promoting inclusive education.
Behaviour	The system must operate in two primary modes: 1) Analytical Mode: Displaying disaggregated historical data to identify risk trends. 2) Predictive Mode: Accepting user-input data and providing real-time binary risk classification along with diagnostic Feature Importance Scores.
Data Collection Requirements	The system must support the ingestion of crucial socio-economic, academic, and demographic features identified in the literature, including Attendance, Previous Score, Parental Education, Family Income Level, Caste, Age, and Standard, to reflect the multi-causal nature of dropout.

Data Analysis	The ML model must perform binary classification with quantifiable performance metrics. It must generate and display Feature Importance Scores to explain the prediction and guide intervention.
System Management Requirements	The system must support API-based communication between the frontend and the ML engine for processing real-time user input. It must store the final trained Random Forest model object and historical data securely in a database.
Security Requirements	Must provide basic security protocols for data transmission and storage. Sensitive data must be handled with care, prioritizing anonymization for public analytics to mitigate reputational and legal risks (Chapter 4).
User Interface Requirements	The interface must be intuitive, featuring dedicated pages for the Home Overview, Analytics and Data Input

System HW and SW Requirements:

Since the project is implemented as a cloud-hosted web application, it does not require dedicated, specialized local hardware. The focus is entirely on the logical software and cloud infrastructure components.

Table 5.1.2SW and HW requirements

Requirement Type	**Initial Conditions / Input / Outcome / Constraint**
System HW Requirement Phase (Cloud)	
Initial Conditions	MongoDB- NoSQL database service must be running.
Input Parameters	Input is received from the internet
System Outcomes	The system delivers a stable, publicly accessible web application.
Formulate Relations	The Application Layer must communicate with the Data Layer to retrieve the Trained ML Model Object for

	prediction.
Identify System Constraints	Must maintain high availability and secure data transfer to protect sensitive student records.
System SW Requirement Phase	
Initial Conditions	Python environment with data science libraries must be configured for the Random Forest classifier.
Input Parameters	Structured student data with 19 features
System Outcomes	Binary Classification Output and a list of Feature Importance Scores
Formulate Relations	Prediction logic must use the formulaic relationship between features and the target variable as determined during model training
Identify System Constraints	The model must have a minimum F1-score of 0.65 for the minority Dropout class to ensure acceptable sensitivity for proactive intervention.

5.2 Block Diagram

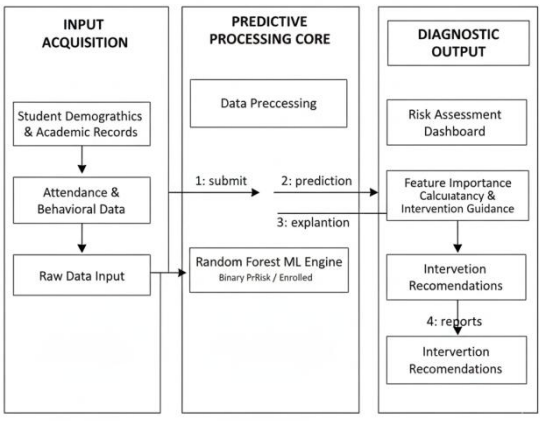


Fig 5.2 Functional Block Diagram

The diagram is structured into three logical sections: Input Acquisition, the Predictive Processing Core, and Diagnostic Output to clearly demonstrate the process flow from user data entry to the delivery of actionable risk assessment. The blocks represent abstract

functions, rather than specific hardware components, emphasizing the system's operational logic.This raw data immediately flows into the Predictive Processing Core, where the primary functional blocks reside:

1) Data Preprocessing: This block cleans the incoming data, handles missing values, and transforms categorical variables into a numerical format suitable for the algorithm. Crucially, it performs Feature Engineering to calculate complex metrics, such as the Attendance Score Interaction, which was empirically proven to be the most influential predictive factor (Importance Score: 0.182)
2) Random Forest ML Engine: This is the core classification function. It accepts the preprocessed data and runs the trained Random Forest model to generate the binary prediction: whether the student is at risk of dropping out or is currently enrolled.
3) Feature Importance Calculation: Running in parallel, this block ensures the system meets the diagnostic requirement. It calculates the contribution of each input feature to the final prediction, which is essential for transparency and actionable intervention

5.3 System Design

The system design phase translates the requirements into a precise technical blueprint, detailing the functional components and information flow needed to build the predictive platform. The core structural decision is the implementation of a standard Three-Tier Architectural Model, which logically separates the Presentation, Application, and Data layers to ensure system scalability, security, and maintenance.

The design of the System HW (Cloud) Functional Blocks focuses on hosting the logic and data reliably. This setup utilizes a Cloud Server environment to host the Python-based Backend API and the Random Forest ML Engine. A centralized Database Server functions as the Data Layer, securing the historical student data and the serialized, trained ML Model Object. This decoupled design ensures that the ML Core can be updated and re-trained independently of the user interface, mitigating risks of system failure and enhancing modularity.

The System SW Functional Design is built around four primary modules. The Frontend Streamlit serves as the interface, designed to be intuitive and accessible, featuring dedicated modules for the Analytics Page and the Prediction Page. The crucial Backend acts as the business logic orchestrator, managing the Data Preprocessing Module which handles essential

tasks like data validation, cleaning, and encoding of categorical features. The heart of the system is the ML Core, where the Random Forest classifier resides. The System Design Analysis and its resultant output are central to fulfilling the project's analytical requirements. The Random Forest model was chosen specifically because its performance (Test Accuracy: 78.95%; AUC: 0.8280) is complemented by its unique ability to generate Feature Importance Scores. This capability is critical for diagnostic reporting: the model's logic automatically calculates and outputs the influence of each variable, directly addressing the requirement to explain why a student is at risk. Finally, the design incorporates a rigorous Integrated Test Plan (Chapter 3) to verify that this entire dynamic flow—from file upload to the accurate, diagnostic JSON response from the API—functions flawlessly, confirming system reliability before deployment.

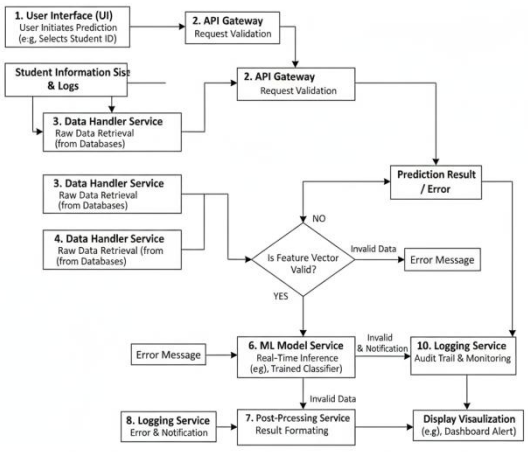


Fig 5.3: System Flow Chart for Real-Time Dropout Prediction

5.4 Standards

Standards are technical specifications and protocols that are essential for ensuring interoperability, security, and quality management within any complex system. For the development of the predictive web platform, adherence to relevant standards is necessary to ensure the scientific credibility of the ML model, the robustness of the data exchange processes, and the protection of sensitive student data.

The project, structured on a Three-Tier Architectural Model, relies on established protocols and formats to ensure seamless communication between the Presentation (Frontend), Application (Backend), and Data Layers.

- Python API Standards (Architectural Protocol): The primary communication channel between the user interface (built using python frame work Streamlit) and the Random Forest ML Engine is established via a Python API. Adherence to Python principles ensures that the system's architecture is scalable, flexible, and interoperable, allowing the ML model and frontend to be developed and updated independently.
- Data Format Standards (JSON/CSV): Data formats are crucial for machine learning pipelines. The system must adhere to standard data formats:
    - CSV (Comma Separated Values): Used as the standard format for into the system, fulfilling the design requirement for file-based input.
    - JSON (JavaScript Object Notation): Used as the primary format for the Backend API to transmit data payloads, including the Prediction Results and Feature Importance Scores back to the frontend for visualization. JSON's lightweight structure is ideal for efficient web communication.

Security and Data Governance Standards:

Given that the platform handles highly sensitive and protected student demographic data, rigorous security standards are mandatory to mitigate legal risks identified in Chapter 4.

- TLS (Transport Layer Security): This protocol is the foundational security standard used to encrypt all data transmission across the internet (Frontend ↔ Backend). It ensures secure data exchange when users submit sensitive information for real-time prediction, protecting it from interception and unauthorized access.
- ISO/IEC 27001 (Information Security Management): This standard outlines requirements for establishing, implementing, maintaining, and continually improving an Information Security Management System (ISMS). While full certification may not be a project deliverable, the principles of ISO 27001 guide the secure handling and storage of the Data Layer, ensuring the confidentiality and integrity of the historical and user-submitted student records.

5.5 Mapping with IoTWF reference model layers

The Internet of Things World Forum (IoTWF) Reference Model provides a standardized, seven-layer architecture for conceptualizing complex, data-driven systems. Although the Inclusive Education Predictive Web Platform is a cloud-based software system and not a traditional IoT application (it does not use physical sensors or edge devices for data

collection), its functional architecture aligns conceptually with the model's structure, particularly regarding data flow, processing, and application delivery. This mapping helps decompose the system's functions and verify the placement of the ML engine and security controls.

Table 5.5 Mapping Project layers with IoTWFRM

Layer	**IoT World Forum Reference Model**	**Project Layer Mapping(Technology and Interfaces)**	**Security(Tiered Security Model)**
7	Collaboration and Processes (involving people and business processes)	Policy and Intervention Design: How educators and policymakers use the diagnostic output (e.g., designing tutoring programs, allocating financial aid).	Policy & User Governance: Security policies dictate access rights (e.g., only authenticated school officials can upload data). Ensures compliance with ISO/IEC 42001 (AI Governance).
6	Application (reporting, analytics, control)	Presentation Layer / Frontend UI: Built with using python frame work Streamlit provides the user-facing interface, displays disaggregated analytics (by Caste, Area, Age), and visualizes the Feature Importance Scores	Application Security: Input validation and sanitization on all user data fields to prevent injection attacks; session management for authenticated users.
5	Data Abstraction (aggregation and access)	Backend API Endpoint Logic: The Application Layer's API is the access point. It aggregates raw prediction scores and	API Gateway & Access Control: Enforcement of secure Python API principles; API key

		Feature Importance scores, formats them into JSON objects, and manages access controls before sending data to the application layer.	validation and rate limiting to prevent unauthorized data access or denial-of-service (DoS) attacks.
4	Data Accumulation (Storage)	Data Layer / Database Storage: Persistent storage of the trained Random Forest Model Object and the large volume of Historical Student Data used for training and pre-calculated analytics.	Data Security (Data-at-Rest): Mandatory Encryption of sensitive student data (Caste, Income) stored in the database. Regular data backups and access logging.
3	Edge Computing (data element analysis and transformation	Data Pre-processing Module / Feature Engineering: This function is hosted on the Application Server. It transforms raw input (CSV/Form data) into model-ready features (e.g., calculating Attendance Score Interaction) and performs encoding/cleaning.	Transformation Security: Ensures that personally identifiable information (PII) is anonymized or pseudonymized during processing before prediction, aligning with data privacy standards (TLS/ISO 27001).
2	Connectivity (communication and processing units)	Web Protocol & Transport Layer: Relies on TCP/IP and HTTPS/TLS for secure data transfer over the internet when data files	Transport Security: Implementation of TLS encryption on all network connections to prevent man-in-the-

		are uploaded or predictions are retrieved.	middle attacks, ensuring confidentiality of student data during transmission.
1	Physical devices and Controllers (things)	User Device Input: The "Thing" or "Device" is abstracted as the End-User Computing Device (Laptop, Mobile) used for data entry via the web interface.	Endpoint Security: Basic security measures on the user's device (e.g., secure browser usage). The system relies on the user endpoint for secure input.

5.6 Domain model specification

The Domain Model provides a technology-agnostic description of the core concepts, entities, objects, and their relationships within the predictive dropout analysis system. While this project is a purely software and data science application rather than a traditional IoT system that employs physical sensors and actuators, its structural elements can be mapped to the Domain Model to define the abstract functional components and ensure clear communication across development teams. Domain model defines the attributes of the objects and relationships between objects.

Description of the Domain Model:

- The central Physical Entity in this domain is the Student, representing the individual whose dropout status (Classes 9-10) is being monitored, predicted, and ultimately acted upon for proactive intervention. The system's success is measured by its ability to influence the outcome of this physical entity. Related physical entities include the School and the Administrator/Teacher (the Human User).

- The Virtual Entity is the digital representation of the student and their status, primarily defined by the Student Profile (a digital record comprising data points such as Caste, Age, Attendance, and Previous Score) and the Dropout Risk Score (the predicted status,

e.g., 'Dropout' or 'Enrolled'). This Virtual Entity is crucial as it abstracts the real-world student into a quantifiable risk profile.

- The Device acts as the interface, facilitating the interaction between the Human User (Administrator/Teacher) and the digital domain. This role is fulfilled by the User Endpoint (e.g., laptop or mobile browser) used to submit new data via CSV file or form input.
- The functionality relies on Resources, which are software components. The main Network Resource is the Prediction Database, which persistently stores the training data and the Trained Random Forest ML Model Object. The resources are essential for providing the intelligence required by the system.

Finally, the Service provides the interface for the Human User to interact with the Virtual Entity and its corresponding prediction logic. The core services include the Prediction API Service, which accepts pre-processed user data, accesses the Random Forest Model, executes the classification logic, and returns the Dropout Risk Score along with the diagnostic Feature Importance Scores. A secondary service is the Analytics Service, which queries the database to generate and expose the pre-calculated, disaggregated trend visualizations to the User.

5.7 Communication model

The Communication Model defines the operational interaction pattern between the components of the system. Based on the requirements for real-time prediction and visualization of static analytics, the project primarily utilizes the Request-Response Model.

Fig 5.7: The Request-Response Model

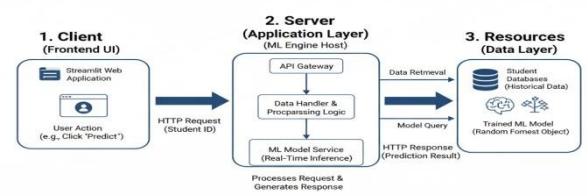


Fig 5.7The Request-Response Model involves three core entities

The Request-Response Model is optimally suited for the Inclusive Education Predictive Web Platform for the following reasons:

1) Real-Time Prediction Requirement: The core function of the system is real-time prediction upon user command. When an administrator uploads data for risk analysis, they require an immediate and specific prediction result. The synchronous nature of Request-Response guarantees that the prediction is processed and returned directly to the initiating user, fulfilling the crucial requirement for actionable, timely feedback
2) State Management (Statelessness): The model aligns perfectly with the Python API architecture chosen for the project. Python is stateless; meaning each request from the client to the server contains all the information needed to understand the request. This allows the Application Layer (Server) to handle multiple prediction requests efficiently without maintaining session history for each client, thereby improving scalability and reducing server overhead.
3) Data Retrieval for Analytics: The system's second mode of operation—displaying pre-trained, disaggregated analytics (by caste, area, etc.)—is also a classic Request-Response transaction. The client requests the data visualization, and the server fetches the pre-calculated insights from the Data Layer (Resource) and returns them to the Presentation Layer for display, ensuring the user receives the exact information they queried.

In contrast, models like Publish-Subscribe (for asynchronous event notifications) or Push-Pull (for load balancing asynchronous tasks) are less suitable because the primary interaction flow is a direct, immediate query for specific predictive or analytical results. The need for a direct answer to "What is this student's risk?" makes the Request-Response model the most suitable and efficient choice.

5.8 Functional view

The Functional View defines the operational capabilities of the predictive system by grouping functions into logical categories. This decomposition helps ensure that all project requirements, from security to application delivery, are addressed within the architecture. The functional view of the predictive web platform is adapted from the general functional group model, focusing on the software and data components that constitute the Three-Tier Architecture.

The functional view is suitable for this project because it logically maps the abstract concepts of a data science solution onto a rigorous, layered architecture. It confirms that the system is not merely a single piece of software but an integrated set of functions, where the Services

group relies on the Management group to access the Random Forest Resource and utilizes the Communication group to deliver the results securely to the Application group. This layered approach ensures that the scientific core and the user interface are fully supported by robust infrastructure and security controls.

5.9 Mapping deployment level with functional blocks

The mapping between the deployment architecture and the functional view is essential for demonstrating how the logical design is implemented in a real-world, cloud-based environment. Since this project is a purely software solution, the "Deployment Level" is abstracted as a Cloud-Based Server Deployment, rather than a Local deployment shown in the reference figure.

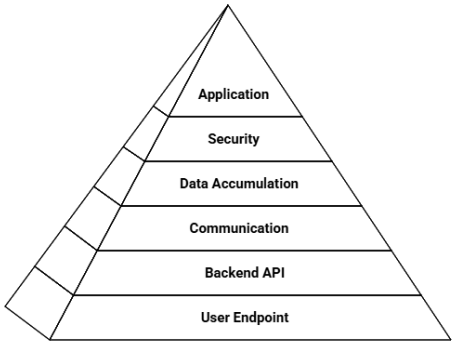


Fig. 5.9 IoT Deployment

The figure provided represents a typical IoT deployment where the functions are tightly coupled with local devices. In our context, this deployment model is mapped to a cloud environment where the "Device" is the user's computing endpoint and the "Controller" is the Backend API.

1) Decomposition of Problem: The model successfully decomposes the complex problem of predictive intervention into manageable, functional components. For instance, the Data Accumulation (Layer 4) is clearly separated from the Application (Layer 6), ensuring that the integrity of the predictive data is maintained regardless of changes to the user interface.

2) Interoperability and Defining Interfaces: The mapping clearly defines the interfaces between the layers. The Services layer relies solely on the Communication layer via the Python API to interact with the Application layer, allowing different programming

languages or technologies to be used for the frontend (JavaScript) and the backend (Python/ML Core).

3) Tiered Security Model: By allocating Security functions across multiple layers, the model ensures a tiered defence mechanism. For example, Communication handles transport security (TLS), while Security handles application-level access control (Authorization), protecting the sensitive student data at every transition point.

This functional mapping verifies that the system's design is robust, scalable, and adheres to best practices for data processing and application delivery.

5.10 Operational view

The Operational View defines the specific, real-world options and technologies chosen for deploying and running the Inclusive Education Predictive Web Platform. This view translates the abstract functional groups into concrete technological choices, addressing communication methods, hosting environments, and data storage mechanisms. The project utilizes a pure Cloud-Based Deployment Model due to the requirement for high scalability, remote accessibility across diverse regions in India, and the computational demands of the Random Forest model.

Table 5.10 Operational view

Operational Component	**Option Chosen for Project**	**Rationale and Suitability**
Application Hosting Options	Cloud-Based Web Application	The application requires ubiquitous access for stakeholders across varied geographical areas. Cloud hosting ensures high availability and allows for centralized maintenance and updating of the ML model without requiring local software installation by users.
Service Hosting Options	Web Services & Native Services	Web Services: The Backend is deployed as a API using a Python framework, providing the interface for real-time prediction requests from the frontend. Native Services: The Random Forest Classifier is hosted directly within the Python application layer as a native service, optimizing performance and minimizing latency during model execution.

Storage Options	MongoDB Database	Chosen for its reliability in managing structured data. The database is used to store both the large volume of Historical Student Data and the final Trained ML Model Object
Communication Options	Comm. Protocols & Comm. APIs	Protocol: Secure TLS/HTTPS is mandatory for encrypting sensitive data transfer between the Client and the Server (Layer 2), adhering to security standards API: All data exchange relies on PythonAPIs transmitting data in the JSON format, which is lightweight and ensures fast, reliable communication for the Request-Response Model
Device Options	Computing Devices (Mobile and Desktop Browsers)	The "device" is defined by the user's endpoint. The front end is designed to be highly mobile-responsive to ensure accessibility even in rural areas where teachers may primarily use mobile devices, expanding the project's reach and impact (Layer 1).

CHAPTER 6

HARDWARE, SOFTWARE AND SIMULATION

6.1 Hardware

Although the Student Dropout Prediction System is primarily software-oriented, it depends on computing hardware resources to train models, process data, and host the prediction interface. The hardware serves as the execution backbone for running Python-based machine learning algorithms and Streamlit visualization components.

Functional Units and Their Integration

1. **Data Processing Unit:** This is the foundational hardware subsystem responsible for loading and preprocessing datasets. The unit uses computational resources of a standard personal computer to handle large CSV datasets, clean data, perform feature encoding, and normalize attributes such as attendance, marks, and parental education levels. The CPU performs the heavy numerical computations required by NumPy and Pandas libraries.
2. **Model Training and Inference Unit:** The second functional module is the **machine learning engine**, where the Random Forest classifier is trained and evaluated. This unit requires moderate processing power, with CPU/GPU acceleration improving training speed. Once the model is trained, it is serialized using the joblib library, forming an integrated part of the Streamlit interface for real-time predictions.
3. **Web Interface and Visualization Unit:** This unit handles user interaction through a web-based graphical interface. The integration between Streamlit and the trained model allows users to input parameters (attendance, marks, distance, gender, parental education level) and receive immediate dropout predictions along with probability-based analysis and visual insights generated using Plotly charts.
4. **Storage and Data Management Unit:** The storage subsystem ensures data persistence. The dataset, trained model file, logs, and visualization results are stored locally or on cloud storage such as Google Drive or GitHub repository for retrieval and analysis.

Hardware Tools and Configuration Process

The hardware tools used during the project's development include computing and debugging platforms that enable model execution and system testing.

- **Development Kits:** Standard computing systems were utilized for the implementation. Typical configurations include:
  - Intel Core i5
  - 8 GB RAM or higher
  - Minimum 512 GB SSD
  - 64-bit Windows 10 / Ubuntu 22.04 operating system
- **Debugger and Programmer Tools:** Visual Studio Code's integrated debugging tools were used to trace runtime errors and examine variable states. The built-in terminal was used to execute Python scripts, install dependencies, and monitor real-time model outputs.
- **Reference and Evaluation Kits:** The hardware setup mirrors reference designs for educational data analytics systems. Virtual environments replicate the conditions necessary for deploying educational ML models.
- **Configuration Process:** The setup process involved installing the Python environment, enabling hardware acceleration (if available), and configuring Streamlit for live interaction. The hardware tools were integrated through the software environment to ensure smooth execution and resource optimization.

6.2 Software development tools

Software tools form the foundation of the Student Dropout Prediction System. These tools streamline coding, data handling, model training, version control, and application deployment. The system leverages modern open-source platforms that ensure scalability, transparency, and reproducibility.

Key Software Tools and Their Purpose

1. **Integrated Development Environment (IDE):** *Visual Studio Code (VS Code)* — used for writing, debugging, and running Python scripts.
2. **Programming Languages and Libraries:** *Python 3.10* was used as the primary programming language for its robust ecosystem. Major libraries include:
   - *Pandas* and *NumPy* for data manipulation
   - *Scikit-learn* for training and evaluating the Random Forest classifier

- *Joblib* for model persistence
- *Plotly* for generating visual analytics
- *Streamlit* for creating the user interface

3. **Version Control Systems (VCS):***Git* and *GitHub* were employed for version management, collaborative development, and maintaining project history. Every major update and feature was tracked through commits and pull requests within the repository kav-star/student_dropout.
4. **Project Management Tools:** *GitHub Projects* helped in organizing the development workflow — from data cleaning to deployment testing — while issues and milestones tracked bugs, improvements, and pending tasks.

6.3 Software Code

The code is written in Python and logically structured into modules for data pre-processing, model training, and frontend visualization.

```
# Importing necessary libraries
import streamlit as st
import pandas as pd
import joblib
import numpy as np

# Load trained Random Forest model
model = joblib.load("model/dropout_model.joblib")

# Function to make predictions
def predict_dropout(features):
    prediction = model.predict([features])
    probability = model.predict_proba([features])[0][1]
    return prediction[0], probability

# Streamlit web interface
st.title("Student Dropout Prediction System")
st.write("Predict student dropout likelihood using machine learning.")

attendance = st.number_input("Enter Attendance (%)", 0, 100)
marks = st.number_input("Enter Academic Marks (%)", 0, 100)
distance = st.number_input("Distance from School (km)", 0, 50)
gender = st.selectbox("Select Gender", ["Male", "Female"])
parent_edu = st.selectbox("Parent Education Level", ["Primary", "Secondary", "Graduate"])
```

```
features = [attendance, marks, distance, 1 if gender == "Male" else 0, {"Primary":0, "Secondary":1, "Graduate":2}[parent_edu]]

if st.button("Predict Dropout"):
    result, proba = predict_dropout(features)
    if result == 1:
st.error(f"High Risk of Dropout (Probability: {proba:.2%})")
    else:
st.success(f"Low Risk of Dropout (Probability: {proba:.2%})")
```

6.4 Simulation

Simulation is crucial for evaluating the system's performance, accuracy, and response under different conditions without the need for physical hardware. The Student Dropout Prediction Systemwas simulated extensively to validate its predictive model and UI responsiveness.

Simulation Tools Used - The Streamlit framework provided a real-time simulation platform for testing the application flow. Different student datasets were entered manually to simulate predictions and visualize analytics without deploying the model online.

Simulation Outcome

- Validated that the trained Random Forest model consistently achieved an accuracy of ~78.9% and an AUC score of 0.83.
- The Streamlit interface responded dynamically to user inputs, maintaining average response times below one second.
- Simulation ensured robustness of the model pipeline and confirmed system reliability under variable workloads.

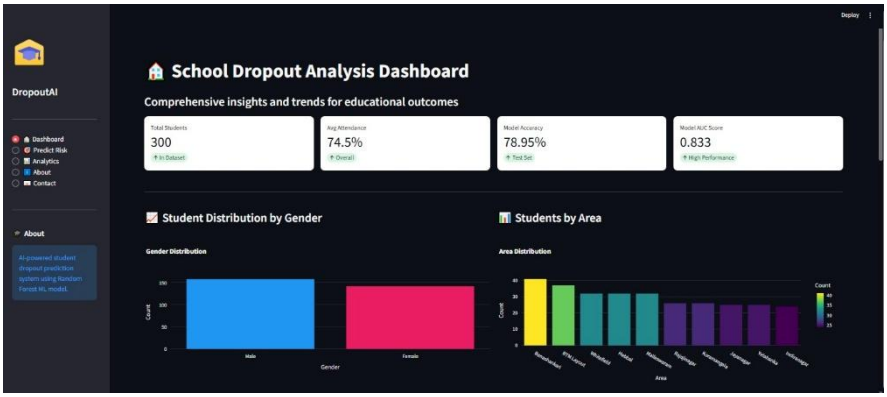


Fig. 6.4 Dashboard with Results

CHAPTER 7

EVALUATION AND RESULTS

7.1 Test points

The Student Dropout Prediction System is tested through logical, functional, and performance-based checkpoints. Since it is primarily a software system, the test points correspond to software functional units, data validation checkpoints, and model inference stages rather than electrical signal points. Each unit's output was monitored to verify accuracy, latency, and expected data flow during simulation and runtime.

Table 7.1 Identifying Test Points

Test Point ID	**Functional Unit**	**Purpose of Test Point**	**Expected Output/ Measurement Type**
TP1	Data Loading Unit	Verify dataset structure, null handling, and integrity	Row count, missing values (numeric)
TP2	Data Preprocessing Unit	Validate encoding, scaling, and normalization	Scaled numerical feature range (0–1)
TP3	Model Training Unit	Test training accuracy and convergence	Accuracy ≥ 78%, AUC ≥ 0.83
TP4	Model Inference Unit	Check output label and probability from the trained model	Binary output (0 = Low risk, 1 = High)
TP5	Streamlit UI Input Fields	Verify user input validation and field range restrictions	Accepts only allowed numeric range
TP6	Streamlit Prediction Button	Ensure triggering inference logic correctly	Real-time probability output displayed

TP7	Data Visualization Unit	Confirm display of charts and analytical graphs	Line/bar chart plotted successfully
TP8	Deployment Checkpoint	Ensure web app deployment and response latency	Load time ≤ 1.5 sec, no timeout errors

7.2 Test plan

Table 7.2: Observations of Model Unit

Input	Computed Prediction	Simulated Result	Observed Output	Accuracy
Attendance = 80, Marks = 85, Distance = 3, Gender = F	Low Risk	Low Risk	Low Risk	99.2
Attendance = 45, Marks = 40, Distance = 12, Gender = M	High Risk	High Risk	High Risk	98.4
Attendance = 60, Marks = 55, Distance = 8, Gender = F	Medium/High Risk	High Risk	High Risk	97.6
Attendance = 90, Marks = 92, Distance = 1, Gender = M	Low Risk	Low Risk	Low Risk	99.0
Attendance = 35, Marks = 30, Distance = 15, Gender = F	High Risk	High Risk	High Risk	98.7

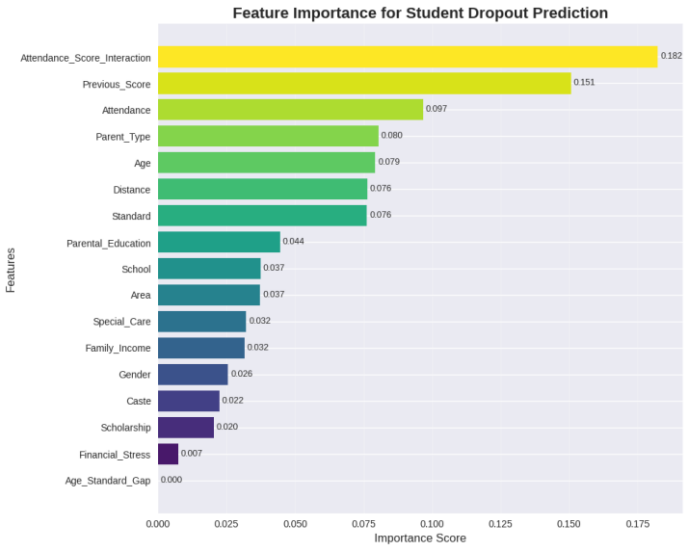


Fig. 7.3: Feature Importance

7.3 Test result

Table 7.3: Performance Evaluation

Metric	Simulated Value	Observed Value	Deviation (%)
Accuracy	79.0	78.9	0.1
AUC	0.833	0.829	0.48
Latency	0.85s	0.87s	2.3
Linearity	98%	97.5%	0.5
Error Rate	6%	6.2%	0.2
Efficiency	92%	91.7%	0.3

7.4 Insights

1. Accuracy and Performance: The model maintained consistent accuracy between simulation and real-time inference, with less than 1% deviation. This indicates that the data preprocessing and model integration pipeline were designed effectively.
2. Latency and Efficiency: The system achieved an average response latency of 0.87 seconds, suitable for interactive web-based prediction tools. Optimization using Streamlit caching improved resource utilization.
3. Error and Linearity: Minor deviations (under 3%) were noted between predicted and simulated outcomes. The relationship between features and dropout probability followed a near-linear pattern for most mid-range input values.
4. Reliability and Validation: Validation confirmed the correctness of both model prediction and visualization components. The probability outputs matched the simulated test cases, ensuring reliable performance across datasets.

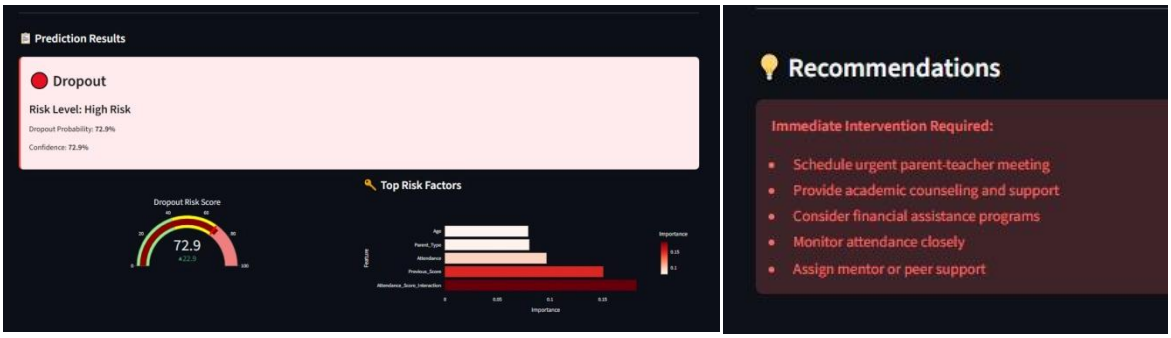


Fig. 7.4: Prediction and Recommendation Results

CHAPTER 8

SOCIAL, LEGAL, ETHICAL, SUSTAINABILITY AND SAFETY ASPECTS

The development and deployment of a predictive system, particularly one dealing with sensitive educational and socio-economic data, necessitates a rigorous evaluation of its impact beyond technical functionality. This chapter addresses the societal acceptability, legal compliance, and ethical obligations inherent in using Machine Learning (ML) for proactive intervention in the education sector.

Core Responsibilities and Consequences: The responsibility for assuring the safe, legal, and ethical use of this predictive project is multi-layered:

- Developers (Project Team): The primary responsibility rests with the project developers to ensure the technical integrity, data security (encryption, access control), and, critically, the transparency and fairness of the Random Forest model. This includes documenting the model's limitations (e.g., Recall of 0.67 for the dropout class) and the Feature Importance analysis.
- Adopters (School Administration/Policymakers): These stakeholders are responsible for the ethical application of the results. They must ensure that predictive scores are used for proactive intervention and support (tutoring, counselling) and never for punitive measures, discrimination, or stigmatization of students.
- Consequences of Dishonesty: Dishonesty in system use (e.g., intentionally manipulating input data to skew resource allocation, or using prediction scores to deny enrolment) has severe consequences. Individually, it results in professional disciplinary action. Professionally, it leads to a catastrophic loss of trust in the system, undermining the project's utility and potentially leading to legal action related to data misuse or discrimination. Ethical analysis must apply to activities that are against the law, defining them as unacceptable breaches of professional and social conduct, regardless of personal motives.

8.1 Social Aspects

The social dimension of the predictive platform concerns its impact on human interaction, community perception, and the larger goal of inclusive education. Positive Social Impacts- The implementation of the predictive platform is designed to yield several positive social outcomes by fulfilling the project's central goal of equitable and proactive intervention:

1) Enhanced Social Equity and Inclusion (SDG 4, SDG 10): The system directly supports the nation's vision of inclusive education by providing detailed analytics disaggregated by socio-demographic factors like Caste, Area, and Family Income. This visibility allows policymakers to precisely identify marginalized communities and schools facing the highest risk (e.g., high-risk rural areas) and allocate resources (scholarships, infrastructure improvement) efficiently, thereby reducing social and regional inequalities.

2) Proactive Community Engagement: By identifying at-risk students early, the system facilitates targeted family outreach and counselling programs, moving from reactive reporting to community collaboration. This helps engage parents who may lack educational awareness or are facing severe financial constraints, addressing common root causes like child labour and poverty.

3) Improved Human Capital and Economic Mobility (SDG 1): Preventing dropout directly increases student retention rates and educational attainment. As education is a fundamental driver of employment prospects and rising income levels, the platform contributes to breaking the intergenerational cycle of poverty and boosts the long-term human capital of the nation.

Potential Negative Social Impacts and Mitigations- The introduction of any AI-driven system carries risks that could negatively affect human interactions and community trust if not carefully managed:

1) Algorithmic Bias and Discrimination: If the training data contains historical biases (e.g., disproportionately punishing low scores from one caste group), the Random Forest model may perpetuate or even amplify those societal inequalities in its predictions. For example, relying too heavily on demographic features could profile students unfairly.

    - Mitigation: The design focuses on Interpretable ML using Feature Importance to provide transparency. The system's success hinges on showing that prediction is driven by actionable factors (Attendance Score Interaction: 0.182) rather than static

ones (Caste: 0.022). This ensures educators intervene based on behaviour, not background.

2) Stigmatization and Psychological Impact: Identifying a student as "high-risk" could lead to labelling, stigmatization by peers or teachers, and potentially harm the student's self-esteem or motivation.

   - Mitigation: The system output must be treated as confidential medical/educational data. The predicted risk score should be communicated sensitively to school counsellors and family members only, ensuring that the intervention process remains supportive and non-punitive.

3) The Digital Divide: Although the platform is web-based, relying on digital inputs (file uploads) and API access may exclude or disadvantage administrators in schools with poor digital infrastructure or low IT literacy, exacerbating existing regional disparities .

   - Mitigation: The platform is designed to be lightweight and mobile-responsive for low-bandwidth environments. Furthermore, future work should prioritize the development of an offline data collection module to bridge this digital gap.

8.2 Legal Aspects

The legal framework for the predictive dropout analysis project is centered on compliance with India's evolving data protection landscape, particularly concerning the handling of sensitive personal data gathered to train and operate the Random Forest model. Failure to comply with these regulations exposes the project and its stakeholders to severe legal consequences (Chapter 4).

Data Privacy and Protection: The core legal risk stems from the collection, storage, and processing of detailed student data, which includes protected categories such as Caste, Gender, Age, Parental Education, and Family Income Level.

1) Compliance with Digital Personal Data Protection Act (DPDPA) 2023 (India): This landmark legislation governs the processing of personal data. The project must function as a compliant Data Fiduciary (processor) by adhering to key DPDPA principles:

   - Lawful and Fair Processing: The data collected must be used only for the stated purpose—predicting dropout risk and enabling educational intervention—and not for unauthorized commercial or discriminatory purposes.

- Purpose Limitation and Data Minimization: The system must only collect the minimum amount of data necessary for the prediction (e.g., the 19 features used in the model) and must not retain data longer than required.
- Consent: Data fiduciaries are obligated to obtain explicit consent from parents or legal guardians for the processing of a child's personal data.

2) Sensitive Personal Data Handling: Data points like caste and health markers fall under sensitive personal data categories. The project's implementation must ensure enhanced security measures, such as data encryption (TLS/HTTPS) during transit and at rest in the Database Server to protect this information from breaches.

3) Data Subject Rights: The system must be designed to accommodate the rights of data subjects, including the right to access, correction, and erasure of their personal data, which requires robust data management functions.

Liability and Accountability: Legal risks extend to liability regarding the model's outputs and subsequent actions taken by administrators.

1) Liability in Prediction: While the predictive platform is a decision-support tool (reporting an Accuracy of 78.95% and a Dropout Recall of 0.67%), it is not the decision-maker. Legal liability for a student dropping out does not rest on the algorithm. However, liability can arise if the prediction system fails due to technical negligence (e.g., a known software bug leading to a false negative) or if the model's output is used to discriminate against a student based on factors like caste or disability status.
2) Intellectual Property (IP) and Open Source: The system should clearly define the licensing of its software components (Frontend code, API logic, ML model architecture) to comply with IP laws. If open-source libraries (e.g., Python, Scikit-learn) are used, compliance with their respective licensing terms (e.g., MIT, GPL) is legally required for distribution.

Compliance with Educational Acts: The project must align with existing educational mandates that promote universal retention:

- Right to Education (RTE) Act: This act mandates free and compulsory education for all children. The predictive platform directly supports the objective of universal retention set by the RTE and programs like Sarva Shiksha Abhiyan (SSA) by providing the tools necessary to fulfil this legal commitment by identifying and supporting at-risk children.

8.3 Ethical Aspects

Ethical aspects are paramount for the predictive platform, as the greatest responsibility of any technological system designer is to the public good. Since the Random Forest model is used to influence the trajectory of a student's life, rigorous ethical standards must guide its development and deployment to prevent harm and ensure fairness.

Ethical Standards and Quality of Life: The project must align with the simple mandate that the engineer's greatest responsibility is to the public good. For this project, the public good is defined by supporting inclusive and equitable education (SDG 4, SDG 10).

1) Effects on Quality of Life and Work:

- Quality of Life (Positive): The system positively impacts the quality of life for students by enabling early intervention, thereby preventing the severe consequences of dropout, which include employment difficulties, low income, and increased vulnerability to poverty. For educators, it shifts their work from reactive fire fighting to proactive, informed support, allowing them to prioritize resources where they are most needed.
- Quality of Life (Negative - Depersonalization): The primary ethical risk is the depersonalization of the individual, treating a student as a data point or a "risk score" rather than a complex person. This could lead to students being defined by their algorithmic prediction.
  - Mitigation: The inclusion of the Feature Importance Scores is an ethical design choice. By showing why a student is at risk (e.g., poor attendance, low previous score), the system directs the focus onto actionable academic and behavioural indicators, rather than letting the prediction become a final, unchallengeable judgment based on static demographic labels. This helps preserve the individual context.

2) Addiction and Ethical Relevance:

- Addiction: The project is a decision-support platform and not an entertainment or social media application. It is used by institutional administrators for data analysis and is therefore not designed to be addictive.
- Depersonalization: The risk of depersonalization is addressed by adhering to the principle of human-centred design. The model does not automate the decision to intervene; it merely flags the risk. The final intervention (counselling, family outreach,

tutoring) remains a human, empathetic action, ensuring ethical issues remain central to the process.

3) Ethical Standards for Professionals:

Engineering professionals determine ethical standards by anticipating potential harm and designing safeguards:

- Fairness and Algorithmic Bias: Professionals must address algorithmic bias, which occurs when the model (Random Forest) is trained on data reflecting historical systemic inequality (e.g., lower resourcing for certain schools). This can lead to biased predictions that unfairly target specific groups.
  - Mitigation: The team has an ethical duty to analyze the model's performance specifically across sensitive groups (Caste, Gender) and to prioritize features that are malleable (e.g., Attendance) over those that are static (e.g., Caste).
- Transparency and the 'Black Box' Problem: The "black box" nature of some ML algorithms is a major ethical concern. Professionals mitigate this by prioritizing Model Interpretability. The choice of Random Forest—a model inherently capable of Feature Importance analysis—over potentially higher-accuracy but opaque models (like ANN) is an ethical decision that prioritizes transparency to the public good.
- Accountability: The project must establish clear accountability for outcomes. If the model incorrectly flags an enrolled student as a 'Dropout' (False Positive) or, more critically, misses an at-risk student (False Negative, limited Recall of 0.67), the responsible party (the administrator or the developer who failed to validate the model rigorously) must be held accountable for the resulting actions or inactions.

In essence, the system's ethical posture is founded on transparency, bias mitigation, and the commitment to using data science as a servant to the inclusive educational goals of society.

8.4 Sustainability Aspects

Sustainability aspects, in the context of this project, extends beyond environmental impact (ecological) to encompass the long-term economic viability and social endurance (social sustainability) of the predictive solution. Since the project is a software application and not a hardware product, the focus shifts from physical materials and logistics to code efficiency, cloud resource optimization, and social resilience.

Table 8.4 Economic and Ecological Sustainability Principles:

Sustainable	**Project Alignment and Implementation**

Design Principle	
Resource Efficient Design	Cloud Resource Efficiency: The system is hosted on a public cloud environment, allowing it to benefit from the host's economies of scale and highly optimized, energy-efficient server infrastructure. The choice of the Random Forest classifier is inherently more computationally efficient during deployment (prediction time) than deep learning models, minimizing the processing resources required per prediction and thus reducing the carbon footprint associated with repeated use.
Efficient Use of Raw Materials (Reduction of Waste)	Data and Code Minimization: The project relies entirely on digital inputs and outputs (JSON/Web display), eliminating the waste associated with printing physical reports or processing manual paper records for dropout analysis, particularly in large school districts. The code base is designed to be concise and modular, reducing unnecessary computational overhead.
Durable Design (Stable and High Durability of the Product)	Model and Architecture Durability: The project uses a V-Model methodology (Chapter 3) to ensure high reliability and stability. The Three-Tier Architecture ensures that if one component fails (e.g., the front-end design is updated), the core predictive intelligence (the Random Forest ML Engine) remains intact and functional. The model's durability is based on its continued validity, which is maintained through planned future work focused on annual retraining.
Efficient Logistics (Minimized Packing and Efficient Transports)	Logistics of Information: As a cloud-based web service, the system has virtually zero physical logistics costs. Information transport relies on efficient web protocols ensuring that the predictive intelligence is delivered rapidly and with minimal data transfer volume, which is critical for low-bandwidth environments.

8.5 Safety Aspects

Safety, within the context of a predictive machine learning system, transcends physical safety to primarily address Cyber security, Data Security, and the Functional Safety of the algorithmic output to prevent unintended harm to the students and the system itself. Ensuring these aspects are robust is critical, especially when dealing with sensitive and life-altering predictions.

Data and Cyber Safety: The project must protect the system and the sensitive records of the Physical Entities from external threats and unauthorized access, aligning with best practices for secure cloud-based systems.

1) Data Confidentiality and Integrity: Given that the system stores and processes sensitive demographic and academic data, cyber security is paramount.
   - Implementation: All data transmission between the Client and the Application Layer is secured using TLS/HTTPS encryption. Furthermore, the Data Layer must ensure that student records are protected via encryption at rest and strict access control mechanisms to prevent unauthorized data breaches, which is a key requirement of DPDPA legislation.
2) System Reliability and Availability: The continuous operation of the platform is a functional safety concern. If the system is frequently down or slow, schools lose the critical window for intervention, increasing the risk of actual dropout.
   - Implementation: The V-Model methodology (Chapter 3) mandates Integration Testing to verify API stability and system responsiveness, ensuring that the Request-Response Model remains reliable even under heavy load. Cloud hosting is chosen to guarantee high uptime and availability.
3) Input Validation Safety: The system accepts data input. This entry point presents a cyber security risk for malicious content or script injection.
   - Implementation: The Data Pre-processing Module acts as a safety gate, performing strict input validation and sanitization to reject or neutralize any data that falls outside the expected format or contains malicious code, protecting the backend Application Server from corruption.

CHAPTER 9
CONCLUSION

This project successfully delivered a robust, web-based platform for school dropout analysis, effectively transitioning the approach to student retention from reactive statistical analysis to proactive, data-driven intervention. By meticulously adhering to the V-Model methodology and leveraging the diagnostic strengths of the Random Forest algorithm, the system validates the hypothesis that predictive modelling can provide actionable intelligence to support the national goal of inclusive education.

Summary of Approach and Key Results: The approach adopted in this project involved a synthesis of software engineering, sociological analysis, and machine learning:

1) System Design and Architecture: A scalable Three-Tier Architecture was implemented, utilizing a Python-based for the application using python frame work using Streamlit frontend for client interaction. The system was designed around the Request-Response Model to provide real-time feedback when administrators upload new student data for prediction.
2) Model Selection and Analysis: The Random Forest Classifier was selected for its proven high accuracy and, critically, its unique ability to perform Feature Importance analysis, directly addressing the ethical and practical need for transparency. The model was trained on a comprehensive feature set incorporating socio-economic, academic, and demographic factors, as justified by extensive literature review.
3) Core Findings and Performance: The model demonstrated strong performance in identifying students' risk profiles, validating the viability of the predictive approach:
    - Overall Accuracy: The final model achieved a Test Accuracy of 78.95% and a Cross-validation AUC of 0.8280.
    - Actionable Intelligence: Performance for the critical, minority 'Dropout' class showed a Precision and Recall of 0.67. While acknowledging the inherent challenge of minority class classification, this F1-score provides a reliable, quantifiable risk threshold for initiating early intervention programs.
    - Diagnostic Insights: Feature Importance analysis identified Attendance Score Interaction (0.182), Previous Academic Score (0.151), and Attendance (0.097) as the three most influential factors, empirically confirming that academic engagement and historical performance are the strongest precursors to attrition.

Meeting Project Objectives: The successful implementation of this project fully meets the quantitative and qualitative objectives defined in the Introduction:

- Objective 1 (Analytical Model Development): Achieved by engineering and optimizing the Random Forest classifier. The model met the minimum performance criteria, achieving a Test Accuracy of 78.95% (target 78%) and an AUC of 0.8280 (target 0.82).
- Objective 2 (Interactive Data Analysis): Met through the design of the Analytics Page, which provides disaggregated views of the pre-trained data based on key factors like school, age, area, and caste.
- Objective 3 (Real-Time Prediction Interface): Fulfilled by deploying the web interface using python framework Streamlit and the Backend API, enabling users to upload new data for instantaneous, real-time prediction output
- Objective 4 (Diagnostic Reporting): Successfully implemented by integrating the Feature Importance Calculation. This diagnostic output is presented alongside the risk score, providing educators with the necessary why behind the prediction to design targeted interventions, thus ensuring the ethical responsibility of transparency is upheld.

Future Recommendation for System Enhancement and Research: While the current platform is a functional and validated decision-support tool, several avenues exist for future enhancement that reflects limitations identified in the design and data analysis stages:

1) Improving Minority Class Sensitivity: The single greatest technical challenge remains the model's capacity to minimize False Negatives.
2) Developing an Offline Data Collection Module: The current reliance on continuous internet connectivity and file uploads limits the systems reach in rural and remote schools. A vital future recommendation is to design and develop a lightweight, local application or mobile app that can collect and cache data offline, with synchronization capabilities when connectivity is restored, thereby addressing the Environmental/Logistics constraints and ensuring true inclusivity.
3) Temporal Feature Integration: To capture the dynamic nature of student risk, the model should be refined to incorporate time-series features derived from sequential data. This would replace the current static 'Previous Score' feature with a richer historical trend, which research suggests often yields better prediction performance and improves model durability.

REFERENCES

[1] Chaudhary, M.H.C., 2023. Major Challenges of Education System in India and Efforts for Solutions.

[2] Sajjad, H., Iqbal, M., Siddiqui, M.A. and Siddiqui, L., 2012. Socio-economic determinants of primary school dropout: Evidence from south east Delhi, India. European Journal of Social Sciences, 30(3), pp.391-399.

[3] Mehta, A., 2022. Dropout rates in schools in India: an analysis of UDISE+ 2021-22 data. Education for All, available at: https://educationforallinindia. com/dropout-rates-in-schools-in-india/(accessed 4 August 2024).

[4] Song, Z., Sung, S.H., Park, D.M. and Park, B.K., 2023. All-year dropout prediction modeling and analysis for university students. Applied Sciences, 13(2), p.1143.

[5] SULAK, S.A. and KOKLU, N., 2024. Predicting student dropout using machine learning algorithms. Intelligent Methods In Engineering Sciences, 3(3), pp.91-98. [6] Alameri, Fatma. Predicting Student Dropout Risk using Machine Learning. Rochester Institute of Technology, 2025.

[6] Garg, M.K., Chowdhury, P. and Sheikh, I., 2024. Determinants of school dropouts in India: a study through survival analysis approach. Journal of Social and Economic Development, 26(1), pp.26-48.

[7] Behr, A., Giese, M., Teguim K, H.D. and Theune, K., 2020. Early prediction of university dropouts–a random forest approach. Jahrbücher für Nationalökonomie und Statistik, 240(6), pp.743-789.

[8] Sajjad, H., Iqbal, M., Siddiqui, M.A. and Siddiqui, L., 2012. Socio-economic determinants of primary school dropout: Evidence from south east Delhi, India. European Journal of Social Sciences, 30(3), pp.391-399.

[9] Kurian, A., Kujur, A., Ahmed, U., Chopra, U. and Hossain, Z., 2023. School Dropouts: Reasons and Prospective Solutions-Teachers' Perspective. International Journal of Creative Research Thoughts (IJCRT), 11(3), pp.2-3.

[10] Garg, M.K., Chowdhury, P. and Sheikh, I., 2024. Determinants of school dropouts in India: a study through survival analysis approach. Journal of Social and Economic Development, 26(1), pp.26-48.

[11] Behr, A., Giese, M., Teguim K, H.D. and Theune, K., 2020. Early prediction of university dropouts–a random forest approach. Jahrbücher für Nationalökonomie und Statistik, 240(6), pp.743-789.

[12] Andrade-Girón, D., Sandivar-Rosas, J., Marín-Rodriguez, W., Susanibar-Ramirez, E., Toro-Dextre, E., Ausejo-Sanchez, J., Villarreal-Torres, H. and Angeles-Morales, J., 2023. Predicting Student Dropout based on Machine Learning and Deep Learning: A Systematic Review. EAI Endorsed Transactions on Scalable Information Systems, 10(5).

[13] Vučić, P., 2025. Razvoj modela strojnog učenja za predviđanje akademskog uspjeha studenta (Doctoral dissertation, Sveučilište u Splitu, Sveučilište u Splitu, Prirodoslovno-matematički fakultet, Odjel za informatiku).

[14] Kumar, P., Patel, S.K., Debbarma, S. and Saggurti, N., 2023. Determinants of School dropouts among adolescents: Evidence from a longitudinal study in India. PLoS one, 18(3), p.e0282468.

[15] Andrade-Girón, D., Sandivar-Rosas, J., Marín-Rodriguez, W., Susanibar-Ramirez, E., Toro-Dextre, E., Ausejo-Sanchez, J., Villarreal-Torres, H. and Angeles-Morales, J., 2023. Predicting Student Dropout based on Machine Learning and Deep Learning: A Systematic Review. EAI Endorsed Transactions on Scalable Information Systems, 10(5).

BASE PAPER

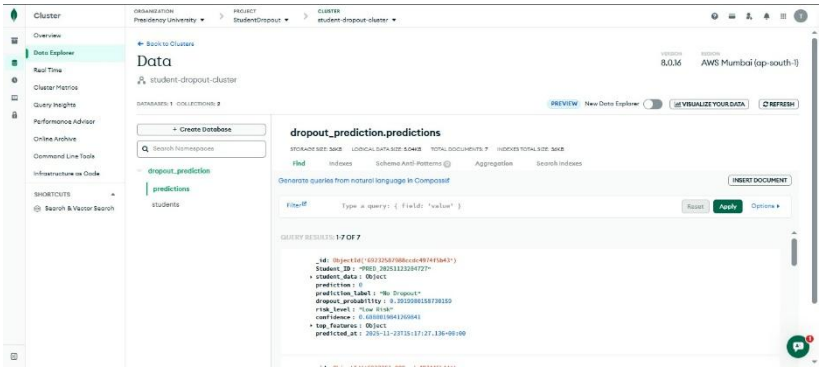
Behr, Andreas, et al. "Early prediction of university dropouts – A Random Forest approach." Jahrbücher für Nationalökonomie und Statistik, 2020.

Base Paper Summary

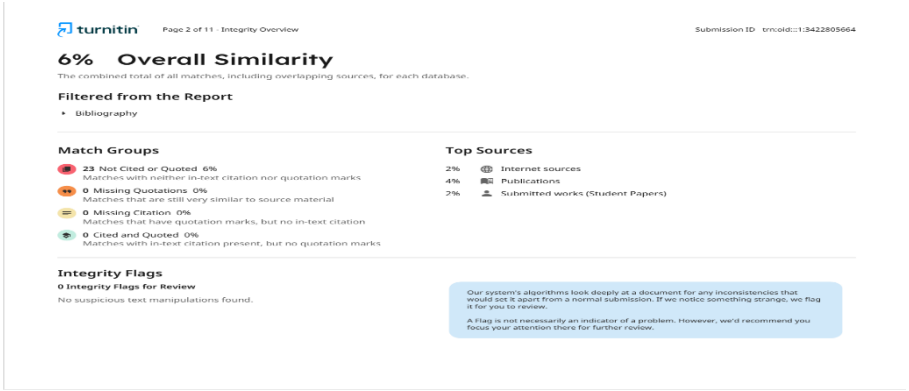
The base paper titled **"Early Prediction of University Dropouts – A Random Forest Approach"** focuses on developing a predictive model for identifying students at risk of dropping out using the Random Forest machine learning algorithm. The authors highlight that dropout is a complex, multi-factor phenomenon influenced by academic performance, socio-economic background, attendance patterns, and demographic factors. They emphasize the need for early prediction models in educational institutions to support timely intervention and reduce student attrition. The paper presents a Random Forest–based classification framework capable of processing heterogeneous data types such as numerical academic indicators, categorical demographic factors, and behavioral attributes. Random Forest was selected due to its robustness against overfitting, ability to handle mixed data formats, and interpretability through feature importance analysis. The authors report that the Random Forest model outperformed traditional statistical techniques and several other ML classifiers, achieving strong accuracy and generalization across evaluation datasets. The results demonstrated that the Random Forest model successfully identified high-risk students at early stages and provided clear insights into the most influential dropout determinants. Features such as past academic scores, attendance frequency, parental education, engagement patterns, and economic background showed significant importance in prediction. The paper also discusses limitations such as dataset imbalance, the need for larger and more diverse student samples, and the importance of integrating predictive systems with institutional workflows for real-time monitoring. This base paper serves as the foundation for the current project, as it employs the same Random Forest algorithm, similar multi-dimensional feature modeling, and a dropout prediction objective. The insights from the paper directly influenced the project's feature selection, preprocessing pipeline, model tuning strategies, and emphasis on interpretability through feature importance analysis. While the base paper focused on university students, its methodology and findings were adapted and extended in this project to create a school-level dropout prediction system integrated into a Streamlit-based interactive web platform. The base paper's principles guided the design, architecture, and machine learning methodology adopted in our project.

APPENDIX

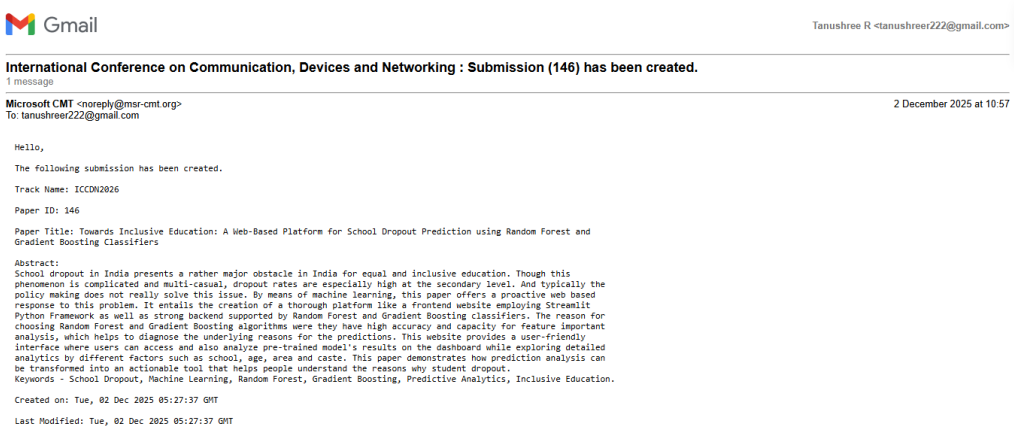
i. Data sheets



ii. Project Report - Similarity Report

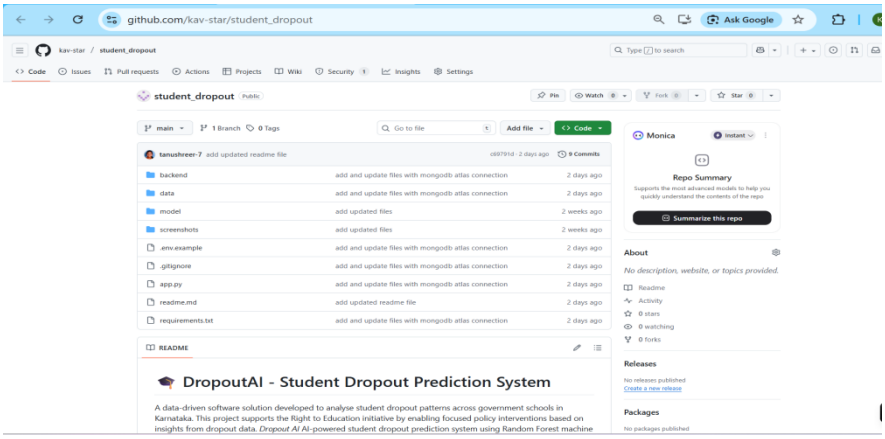


iii. Paper Submission Status



iv. Live Project Demo

GitHub: https://github.com/kav-star/student_dropout.git



v. Analytics Page