IJCRT.ORG ISSN : 2320-2882
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Out Come Based Hands-On Learning Model For Programing Education

Sandeep Padesur, Master of Science
Software Engineer Faculty
Cranes Varsity Private Ltd.
Bengaluru, Karnataka, India

Abstract: Programming education in technical institutions often faces challenges related to student engagement, logical thinking, and consistent learning outcomes. Traditional lecture-based delivery restricts skill development and may not adequately prepare learners for real-world problem-solving requirements. This research paper proposes an outcome-based learning model integrating hands-on problem solving and Learning Management System (LMS) tracking to improve student performance in programming, data structures, and software development subjects.

The study is based on real training data collected from 6,000 engineering students trained across multiple colleges and corporate environments. Student learning patterns, assessment scores, coding outputs, completion rates, and LMS activity logs were analysed to evaluate skill progression. Results indicate measurable improvement in programming efficiency, logical reasoning, and concept retention. Learners who engaged consistently in LMS-driven learning and structured lab-based activities demonstrated significantly higher performance.

This model demonstrates the value of integrating digital learning platforms, personalized feedback, and practical coding sessions in computing education. It supports the shift from theory-based instruction to applied learning environments.

1 Introduction:

Programing education plays a foundational role in developing skilled software professionals and computer science graduates. However, traditional lecture-oriented instruction frequently results in limited student involvement and insufficient hands-on exposure. Students often understand syntax but struggle to apply logic in real-world coding situations, leading to reduced problem-solving confidence and lower retention. Outcome-Based Education (OBE) offers a structured system that aligns teaching objectives, learning outcomes, instructional activities, and assessment systems. OBE focuses on measurable skill development, enabling students to demonstrate competency rather than passive knowledge retention.
Hands-on learning approaches reinforce conceptual understanding by immersing students in real coding practice. Through repeated exposure to programming exercises and algorithmic challenges, learners exhibit improved execution accuracy, code readability, and debugging efficiency.
Learning Management Systems (LMS) are redefining the academic learning process, enabling digital tracking, automated assessments, progress visualisation, and self-paced learning opportunities. LMS analytics allow instructors to identify learning gaps and personalise guidance.
This research evaluates the combined impact of outcome-based instruction, hands-on programming practice, and LMS usage on student performance.

2 LITERATURE REVIEW

Studies across computer science and education research consistently show that learners benefit from practical coding exposure. Rao & Kumar (2020) found that students engaged in frequent coding activities demonstrated marked improvement in logic and execution accuracy.
Spady (1994) established the foundation for OBE principles, highlighting that student achievement improves when learning goals are transparent, measurable, and activity driven. Spady observed increased retention and improved long-term outcomes in skill-based subjects.
Patel (2022) reported that LMS-based assessments produce higher engagement and self-directed learning. Students using LMS platforms were more aware of their learning progress, error frequency, and performance trends.
Although individual components—hands-on learning, LMS tracking, and outcome-based design—have been independently studied, limited work addresses their combined effect. This study fills that gap by integrating and evaluating all three components within a unified instructional model.

3. METHODOLOGY

3.1 Participants

Data was collected from a sample of 6,000 students from engineering colleges, degree courses, and corporate training environments. Participants varied in skill level, ranging from beginners to experienced learners.

3.2 Subjects Trained

- C Programming
- C++ Programming
- Data Structures
- Java
- Python

3.3 Research Structure

The research followed a three-stage process:
Stage 1: Pre-training assessment
Stage 2: Ongoing LMS activity analysis
Stage 3: Post-training evaluation

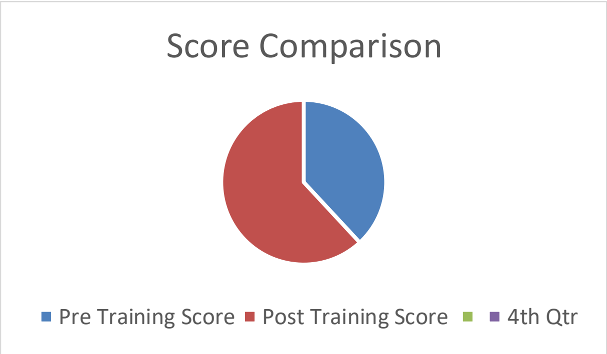
3.4 Performance Metrics

Learning improvement was measured using program execution accuracy, logical reasoning development, syntax and logical error frequency, LMS completion rates, assessment scores, and retention clarity.

4. DATA ANALYSIS AND RESULTS

4.1 Score Comparison

Pre-Training Score: 48%
Post-Training Score: 78%
→ Improvement: +30%



4.2 Execution Accuracy

Execution Success Rate increased from 52% to 90%.
Syntax and logical errors reduced significantly.

4.3 LMS Usage Outcomes

High LMS Usage Group: 84% average score
Medium LMS Usage Group: 74%
Low LMS Usage Group: 63%

4.4 Subject-Wise Retention

C Programming: +32%
C++: +35%
Data Structures: +42%
Java: +37%
Python: +40%
These results confirm measurable improvement across programming domains.

5. DISCUSSION

The study proves that integrating hands-on programming practice with outcome-based instruction results in deeper conceptual development and stronger performance. Students demonstrated increased problem-solving capability, coding accuracy, and execution speed.

The LMS data correlation confirms that consistent digital engagement improves retention and skill clarity. High LMS users achieved significantly better academic results, indicating that structured digital learning pathways enhance focus and accountability.
When compared to traditional lecture formats, the blended OBE-LMS-hands-on approach created meaningful learning experiences and built long-term programming confidence.

6. CONCLUSION AND FUTURE SCOPE

The integrated learning model successfully improved student outcomes in programming subjects. Learners recorded significant increases in accuracy, retention, and logical reasoning. The combined strategy of outcome-driven course planning, LMS progress tracking, and active programming practice was highly effective in enhancing learning quality.
Future work may include applying the model to advanced computing domains, studying long-term retention outcomes, expanding LMS tracking analytics, and evaluating industry-level skills and employability.

ACKNOWLEDGEMENT

I extend my sincere appreciation to Cranes Varsity Private Ltd. for offering a highly supportive and collaborative environment during the course of this research. The institutional guidance, training exposure, and technical infrastructure enabled the successful execution of this study. I am also grateful to my colleagues and students for their cooperation, engagement, and meaningful contributions, which added significant value to the overall outcome of this work.

REFERENCES

[1] W. Spady, “Outcome-Based Education: Critical Issues and Answers,” 1994.
[2] A. Rao and R. Kumar, “Effectiveness of practical programming activities,” 2020.
[3] P. Srivastava, “Hands-on learning approaches for improving programming logic,” 2021.
[4] H. Huang, “Comparative efficiency of programming pedagogies,” 2020.
[5] K. Patel, “LMS-driven coding efficiency in programming education,” 2022.