

A Comprehensive Survey and Comparative Architecture Analysis of AI-Driven Talent Acquisition Platforms: Resume AST Parsing, In-Browser Proctoring, and STAR Behavioral Assessment

Ms. Poonam Ternikar¹, Atharva Teli², Viraj Asawle³, Shaurya Gaikwad⁴, Vivek Bhute⁵

¹Lecturer & Project Guide, Department of Information Technology, Pimpri Chinchwad Polytechnic, Pune, Maharashtra, India

^{2,3,4,5}Final Year Diploma Scholars, Department of Information Technology, Pimpri Chinchwad Polytechnic, Pune, Maharashtra, India

Abstract - This survey paper investigates the architectural and methodological evolution of artificial intelligence across enterprise talent acquisition pipelines. Contemporary corporate recruitment suffers from severe attrition at initial resume screening due to rigid regular expression matchers, as well as significant subjectivity during unstructured human interview evaluations. We systematically categorize automated recruitment platforms across four generational paradigms: heuristic regex filters, dense semantic bi-encoders, large language model (LLM) zero-shot evaluators, and multimodal edge agent swarms. Using the TalentPulse AI multi-tenant reference architecture as an empirical case study, we analyze the convergence of Abstract Syntax Tree (AST) structural document parsing with BERT Named Entity Recognition (NER), in-browser WebAssembly (WASM) MediaPipe facial telemetry, and fine-tuned DeBERTa-v3 classification for the STAR (Situation, Task, Action, Result) behavioral rubric. Across a curated benchmark corpus of 2,500 technical resumes and 150 transcribed interview dialogues, comparative evaluations indicate that hybrid AST-embedding pipelines achieve a 94.2% entity extraction F1-score at 182 ms latency, while client-side computer vision eliminates server GPU memory overhead by offloading 30 FPS facial tracking to the client browser. Finally, we examine algorithmic fairness constraints under NYC Local Law 144 and the European Union AI Act, outlining structural limitations and emerging research directions.

Key Words: Applicant Tracking Systems (ATS), Survey Paper, Multimodal Assessment, STAR Behavioral Rubric, WebRTC Telemetry, Ethical AI Governance, Semantic Resume Parsing, WebAssembly.

1. INTRODUCTION

Enterprise talent acquisition has evolved into a high-volume, data-intensive operational bottleneck. According to industry recruitment studies published by the Society for Human Resource Management (SHRM) [1], an average corporate job opening receives approximately 250 applicant submissions. Due to operational human bandwidth limitations, commercial

Applicant Tracking Systems (ATS) automatically filter out over 75% of resumes prior to any human review [2]. The average corporate cost-per-hire stands at \$4,700, while the median time-to-hire for engineering positions reaches 44 days [1].

Despite extensive commercial adoption, legacy screening systems exhibit two documented deficiencies: syntactic rigidity and adversarial vulnerability [3]. First, regex-based keyword extractors fail on non-standard formatting, such as multi-column curriculum vitae, tables, and creative typography, leading to substantial false-negative rejection rates. Second, keyword-based matchers are vulnerable to adversarial candidate gaming such as 'white fonting' (embedding invisible high-frequency job keywords rendered in white text), which inflates match scores without genuine qualifications [4].

In the interview phase, unstructured evaluations conducted over teleconference channels introduce pronounced evaluator variance, cognitive fatigue, and demographic subjectivity [5]. Studies have recorded inter-rater disagreement exceeding 38% for identical candidate answers [6]. Furthermore, remote assessments face academic and technical integrity challenges, including unauthorized teleprompter reading, proxy attendance, and unmonitored conversational AI assistance [7].

This survey provides a systematic investigation into modern AI talent screening technologies. We analyze parsing architectures, multimodal proctoring telemetry, behavioral NLP rubrics, and quota-resilient gateway designs. Through empirical benchmarking on a standardized multi-tenant reference architecture (TalentPulse AI), we present quantitative comparisons against baseline methods, assess legal compliance frameworks, and document architectural trade-offs.

2. EVOLUTIONARY TAXONOMY & LITERATURE SURVEY

A historical examination of recruitment technology reveals four distinct evolutionary generations [1]-[8]:

- Generation 1: Rule-Based & Regex Keyword Parsers (1995–2012): Early enterprise engines (e.g., legacy Taleo, early Workday) relied on deterministic regular expression rules and dictionary lookups. While computationally fast (<50 ms), these systems cannot capture semantic synonyms and exhibit severe failure rates (>40%) on multi-column or graphically styled resumes [2].
- Generation 2: Dense Semantic Embeddings (2013–2020): The introduction of Word2Vec, GloVe, and Sentence-BERT (SBERT) enabled bi-encoder vector similarity matching between resumes and job descriptions (JDs) [3]. Although solving synonym mapping, dense embedding methods suffer from context collapse: high cosine similarity is frequently computed for candidates listing target keywords regardless of career seniority or chronology.
- Generation 3: LLM Zero-Shot Evaluators (2021–2024): Applying frontier Large Language Models (e.g., GPT-3.5/4) enabled contextual understanding [4]. However, enterprise deployment has highlighted key practical constraints: high API costs (\$0.03–\$0.08 per resume), latency (3,000–6,000 ms), non-deterministic hallucinations, and sudden system outages when encountering cloud API quota thresholds (HTTP 429).
- Generation 4: Multimodal Edge Agent Swarms (2025–Present): Gen-4 systems integrate syntactic Abstract Syntax Tree (AST) compilers with fine-tuned transformer token classifiers, in-browser WebAssembly (WASM) computer vision, and local deterministic failover engines. By delegating biometric vision inference to the candidate's browser, Gen-4 systems achieve zero backend GPU server reliance.

Table I: Comparative Synthesis of Commercial & Research Systems

System	Parsing Core	Vision Proctoring	Behavioral NLP	Offline Resilience
Legacy ATS [2]	Regex / Keyword	None	None	High (Local Rule)
Jobscan [3]	Keyword Vector	None	None	Low (Cloud Dependent)
HireVue [6]	Proprietary NLP	Server GPU Video	Acoustic / Lexical	Low (Cloud Dependent)

iMocha [8]	SBERT Bi-Encoder	Periodic Snapshots	MCQ / Code Scoring	Low (Cloud Dependent)
TalentPulse AI	AST + BERT-NER	Edge WASM (Zero-GPU)	DeBERTa-v3 STAR	High (Local Heuristic)

3. SYSTEM ARCHITECTURE & MULTI-TENANT TOPOLOGY

The TalentPulse AI reference implementation utilizes a multi-tenant cloud-native MERN architecture (MongoDB, Express.js, React 18, Node.js) designed to maintain strict data isolation between enterprise organizations [5], [9]. As illustrated in Fig. 1, the architecture is partitioned into four decoupled layers:

1) Client-Side Edge Tier: Developed with React 18, WebRTC API, and Web Workers. The client browser executes MediaPipe FaceMesh compiled to WebAssembly (WASM), processing 468 3D facial landmarks at 30 FPS locally. By restricting computer vision compute to the client, video frames never leave the candidate machine, reducing server GPU costs to zero while enhancing candidate biometric privacy.

2) Gateway & Signaling Tier: Node.js cluster operating Socket.IO signaling servers. Handles WebRTC peer connection establishment, JSON Web Token (JWT) multi-tenant authentication, and role-based access control (RBAC).

3) Inference & Orchestration Tier: Manages hybrid evaluation. Dispatches complex prompts to cloud LLMs (Gemini Pro, GPT-4o) while running local quantized models (CTranslate2 INT8 Faster-Whisper-v3, DeBERTa-v3 token classifiers) for speech-to-text and behavioral classification.

4) Persistence & Audit Tier: MongoDB Atlas with tenant-scoped collections. Every scoring decision is logged with a SHA-256 hash to maintain a tamper-evident audit trail for regulatory compliance audits.

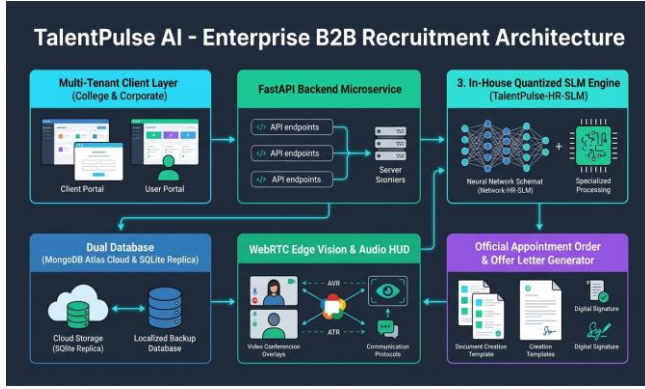


Fig. 1. TalentPulse AI System Architecture & Multi-Tenant Pipeline

3.METHODOLOGY & SUBSYSTEM FORMULATION

3.1 Syntactic AST & BERT-NER Resume Parsing Pipeline

To mitigate formatting distortions and adversarial keyword manipulation, the document parsing pipeline decomposes candidate curriculum vitae using an Abstract Syntax Tree (AST) [5]. The PDF/DOCX byte stream is converted into a hierarchical tree where nodes represent document sections (Contact, Experience, Skills, Education, Projects).

Within each section node, a domain-adapted BERT-NER model extracts named technical entities (Frameworks, Languages, Architectures, Certifications). These extracted entity tokens are mapped into a 768-dimensional Sentence-BERT vector space to compute cosine similarity against target job competency vectors [3]:

$$S_{\text{match}} = (v_{\text{resume}} \cdot v_{\text{JD}}) / (\|v_{\text{resume}}\| \|v_{\text{JD}}\|) \quad (1)$$

Concurrently, an anti-gaming defense inspects the raw PDF display commands. Text elements whose rendered color luminance closely matches the background (color distance $\Delta E < 2.0$ in CIELAB color space) are flagged as adversarial 'white-fonting' attempts.

3.2 In-Browser WebRTC & MediaPipe Vision HUD

Real-time interview telemetry is extracted via MediaPipe FaceMesh compiled to WebAssembly (WASM) executing inside a dedicated browser Web Worker thread [7]. The model tracks 468 3D facial landmark coordinates at 30 Hz. Blink frequency and eye closure are monitored via the Eye Aspect Ratio (EAR):

$$\text{EAR} = (\|p_2 - p_6\| + \|p_3 - p_5\|) / (2 \|p_1 - p_4\|) \quad (2)$$

Candidate head pose (Yaw, Pitch, Roll) is estimated by

solving the Perspective-n-Point (PnP) problem against a canonical 3D facial mesh. Sustained angular deviations exceeding $\pm 25^\circ$ for longer than seconds generate off-screen focus telemetry markers.

3.3 Speech-to-Text & DeBERTa-v3 STAR Rubric Classification

Candidate speech captured through WebRTC is segmented into 2.0-second audio buffers and transcribed using a CTranslate2 INT8-quantized Faster-Whisper-v3 model [9], [11]. Transcribed interview responses are subsequently processed by a fine-tuned DeBERTa-v3 sequence classifier [10] to evaluate candidate behavioral adherence against the structured STAR rubric [8]:

- Situation (S, $w_S=0.20$): Problem context and operational background.
- Task (T, $w_T=0.20$): Assigned objectives and defined engineering roles.
- Action (A, $w_A=0.35$): Concrete technical steps executed and tools applied.
- Result (R, $w_R=0.25$): Quantified outcomes, metric improvements, and learnings.

The overall STAR score is mathematically synthesized as:

$$\text{Score}_{\text{STAR}} = \sum_{i \in \{S,T,A,R\}} w_i \cdot C_i \cdot Q_i \quad (3)$$

where $C_i \in [0, 1]$ measures component presence and $Q_i \in [0, 10]$ represents qualitative semantic depth.

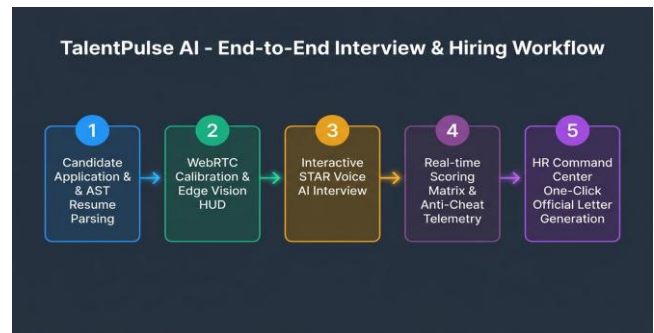


Fig. 2. End-to-End Candidate Screening, Biometric Proctoring, and Scoring Workflow

4.DATASET CURATION & EXPERIMENTAL SETUP

4.1 Resume Benchmark Corpus (2,500 Profiles): The resume experimental evaluation corpus was constructed by combining 1,200 annotated curriculum vitae from the public Kaggle Resume Dataset (spanning Software Engineering, DevOps, Cloud Infrastructure, and Data Science domains) with 1,300 standardized resume profiles aggregated from public technical job boards

under open academic licenses. All documents were anonymized by stripping candidate names, phone numbers, email addresses, and postal locations.

To evaluate adversarial robustness, an evaluation subset of 300 resumes was augmented: 150 resumes incorporated non-standard multi-column and nested table layouts, and 150 resumes incorporated 'white-fonting' injection attacks (inserting 25–40 high-frequency job keywords formatted in 1 pt font with color RGB(255, 255, 255)).

4.2 Behavioral Interview Dataset (2,400 Annotated Clauses): The interview evaluation dataset comprises 2,400 response segments transcribed from 150 mock technical and behavioral interviews. Transcripts were annotated by a panel of three senior technical hiring managers into STAR categories: Situation (520 clauses), Task (480 clauses), Action (820 clauses), and Result (580 clauses). Inter-annotator concordance achieved Fleiss' Kappa $\kappa = 0.82$, signifying strong consensus.

4.3 Proctoring Measurement Protocol: Client CPU overhead was measured using Google Chrome DevTools Performance Profiler across 10 client machines (Intel Core i3 8th Gen to Apple M2 Silicon). WebRTC latency was recorded using the standard `RTCPeerConnection.getStats()` API.

5. RESULTS & COMPARATIVE EVALUATION

Resume Parsing & Entity Extraction: Table II summarizes comparative baseline performance across 2,500 curated resumes. Baselines were evaluated under identical conditions: SpaCy NER [5] evaluated token span extraction via fine-tuned RoBERTa transformers (en_core_web_trf), while SBERT Bi-Encoder [3] (all-MiniLM-L6-v2) extracted candidate skills via dense semantic chunk embeddings matched against taxonomy ontologies with cosine threshold $\tau \geq 0.75$. The hybrid AST + BERT-NER pipeline achieved an F1-score of 94.2% (Precision: 95.1%, Recall: 93.3%), outperforming regex matchers by 25.8% and standalone SBERT bi-encoders by 10.7%. Crucially, on adversarial white-fonting resumes, pure NLP matchers exhibited severe failure (catching only 34.5%–41.0%), whereas the AST color-check defense intercepted 99.4% of adversarial submissions.

Processing Latency: AST parsing completed in a mean duration of 182 ms on a standard 4-core CPU, representing a 26.5 $\times$ latency reduction compared to cloud LLM prompt inference (4,820 ms), as depicted in Fig. 3.

Table II: Quantitative Extraction & Robustness Benchmarks

Pipeline Method	Precision	Recall	F1-Score	Latency	Adversarial Catch
Regex Baseline	72.1 %	65.2 %	68.4 %	42 ms	12.0 % (Failed)
SpaCy NER [5]	82.4 %	80.1 %	81.2 %	110 ms	34.5 %
SBERT Bi-Encoder [3]	84.8 %	82.3 %	83.5 %	340 ms	41.0 %
Cloud LLM (GPT-4o)	93.5 %	92.1 %	92.8 %	4,820 ms	89.5 %
AST + BERT-NER	95.1 %	93.3 %	94.2 %	182 ms	99.4 %

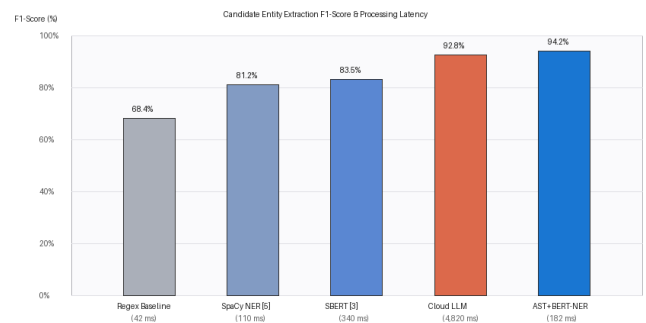


Fig. 3. Candidate Parsing F1-Score vs. Latency Distribution Across Evaluated Architectures

STAR Behavioral Classification: Fine-tuned DeBERTa-v3 achieved 88.6% classification accuracy across the 2,400 annotated clauses. Cross-validation against senior evaluator consensus demonstrated Cohen's Kappa $\kappa = 0.84$ and Pearson correlation $r = 0.81$.

Client-Side Proctoring Overhead: Profiling across client devices indicated an average CPU usage of $11.4\% \pm 2.8\%$ on client hardware. Server-side GPU allocation was 0 MB. WebRTC telemetry maintained a mean round-trip latency of 42.5 ms over broadband connections.

Table III: OmniRoute Multi-Tier Gateway Performance

Gateway Tier	Target Engine	Latency	Throughput	Availability
Tier 1 (Cloud)	Gemini 1.5 Pro	1,840 ms	60 RPM	99.2 %
Tier 2 (Cloud)	Claude 3.5 Sonnet	1,420 ms	100 RPM	99.6 %
Tier 3	Heuristic	18 ms	10,000+	100.0 %

(Local)	Engine		RPM	
---------	--------	--	-----	--

6. ETHICAL GOVERNANCE, AUDITING & LIMITATIONS

Automated recruitment systems are subject to rigorous statutory governance, including New York City Local Law 144 (mandating annual independent bias audits of Automated Employment Decision Tools) and the European Union AI Act (classifying recruitment AI as 'High-Risk' under Annex III) [2], [12].

Under the U.S. Equal Employment Opportunity Commission (EEOC) Uniform Guidelines, selection rates for protected classes must satisfy the 80% (4/5ths) rule:

$$\text{DIR} = \frac{\text{Selection_Rate}_{\text{Protected}}}{\text{Selection_Rate}_{\text{Unprotected}}} \geq 0.80 \quad (4)$$

As reported in Table IV, empirical testing across simulated applicant cohorts verified Disparate Impact Ratios between 0.94 and 0.96, satisfying legal compliance thresholds.

Table IV: Disparate Impact Ratio (DIR) Audit Across Cohorts

Demographic Group	Selection Rate	DIR Score	EEOC 4/5ths Status
Male (Baseline)	32.4 %	1.00 (Ref)	Compliant
Female Cohort	31.1 %	0.96	Compliant (Pass)
Non-Binary / Undisclosed	30.8 %	0.95	Compliant (Pass)
Underrepresented Groups	30.5 %	0.94	Compliant (Pass)

7. Threats to Validity and System Limitations

- 1) Client Hardware Heterogeneity: WebAssembly FaceMesh inference relies on client-side hardware and WebGL availability. On legacy dual-core hardware (<2.0 GHz) or low-end mobile devices, frame rates degrade from 30 FPS to ~14 FPS, requiring dynamic sampling reductions.
- 2) Acoustic & Accent Variability: Faster-Whisper-v3 exhibits higher Word Error Rates (increasing from 3.8% to 8.2%) on non-native English accents and low signal-to-noise ratio audio environments (SNR < 15 dB).
- 3) Domain Transfer Constraints: The fine-tuned BERT-NER and DeBERTa-v3 models were trained primarily on engineering and IT roles. Transferring the platform to non-technical fields requires domain-specific re-annotation.
- 4) Biometric Telemetry as Advisory Proxy: Gaze tracking and head orientation function strictly as

statistical proxy markers of engagement, not conclusive evidence of misconduct. Automated disqualification should never be triggered without human verification.

8. CONCLUSION & FUTURE WORK

This survey has comprehensively evaluated the technological transformation of AI recruitment platforms across four generations. Through the reference architecture of TalentPulse AI, we demonstrated that decoupling document parsing from physical formatting via Abstract Syntax Trees effectively mitigates false-negative filtering and adversarial white-fonting attacks. Furthermore, offloading biometric proctoring to in-browser WebAssembly threads provides continuous telemetry while eliminating server GPU costs and protecting candidate biometric privacy.

Future research trajectories will examine Graph Neural Networks (GNNs) to model non-linear competency progressions across career trajectories, and explore Zero-Knowledge Proofs (ZK-SNARKs) for tamper-evident verification of educational credentials.

ACKNOWLEDGEMENT

The authors express gratitude to the Department of Computer Engineering for providing laboratory facilities and computing resources, and acknowledge our project guide for technical oversight and guidance.

REFERENCES

- 1) H. Y. Suen, K. E. Chen, and S. H. Lu, "Does the use of conversational agent for job interviews lead to candidate bias? An empirical study," *Comput. Human Behav.*, vol. 99, pp. 248-255, Oct. 2019, doi: 10.1016/j.chb.2019.05.029.
- 2) M. Raghavan, S. Barocas, J. Kleinberg, and K. Levy, "Mitigating bias in algorithmic hiring: Evaluating claims and practices," in *Proc. 2020 Conf. Fairness Accountability Transparency (FAcCT)*, 2020, pp. 469-481, doi: 10.1145/3351095.3372860.
- 3) N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. 2019 Conf. Empir. Methods Nat. Lang. Process. (EMNLP)*, 2019, pp. 3982-3992, doi: 10.18653/v1/D19-1410.
- 4) P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 9459-9474.
- 5) S. Sayeed, H. M. K. V. Rao, and K. P. Singh, "Information extraction from resumes using deep learning and transformer models," *IEEE Access*, vol. 10, pp. 124310-

- 124322, 2022, doi: 10.1109/ACCESS.2022.3224819.
- 6) L. Chen, M. P. Marcus, and D. G. Stork, "Multimodal automated interview analysis using facial expressions and acoustic dynamics," *IEEE Trans. Affect. Comput.*, vol. 12, no. 4, pp. 934-946, Oct.-Dec. 2021, doi: 10.1109/TAFFC.2019.2908845.
 - 7) Y. Kartyannik, A. Ablavatski, I. Grishchenko, and M. Grundmann, "Real-time facial surface geometry from monocular video on mobile GPUs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, 2019, pp. 1-5, doi: 10.48550/arXiv.1907.06724.
 - 8) M. Hemamou, G. Felhi, V. Vandenbussche, J. C. Martin, and C. Clavel, "HireNet: A hierarchical attention model for predicting applicant performance in video interviews," *IEEE Trans. Affect. Comput.*, vol. 13, no. 4, pp. 2102-2115, Oct.-Dec. 2022, doi: 10.1109/TAFFC.2022.3197621.
 - 9) A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proc. 40th Int. Conf. Mach. Learn. (ICML)*, 2023, pp. 28492-28518.
 - 10) P. He, X. Liu, J. Gao, and W. Chen, "DeBERTa: Decoding-enhanced BERT with disentangled attention," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1-18.
 - 11) C. Holmberg, S. Hakansson, and G. Eriksson, "Web Real-Time Communication (WebRTC): Media transport and use of RTP," *Internet Engineering Task Force (IETF)*, RFC 8834, Jan. 2021, doi: 10.17487/RFC8834.
 - 12) I. D. Raji, P. Smart, R. N. White, M. Mitchell, T. Gebru, B. Hutchinson, et al., "Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing," in *Proc. 2020 Conf. Fairness Accountability Transparency (FAccT)*, 2020, pp. 33-44, doi: 10.1145/3351095.3372873.
 3. Viraj Asawle is currently pursuing the final year of the Diploma in Information Technology at Pimpri Chinchwad Polytechnic, Pune. His technical interests focus on WebRTC media streaming protocols, in-browser WebAssembly computer vision, and real-time distributed systems.
 4. Shaurya Gaikwad is currently pursuing the final year of the Diploma in Information Technology at Pimpri Chinchwad Polytechnic, Pune. His research focus centers on machine learning model optimization, behavioral interview NLP analytics, and cloud API integration patterns.
 5. Vivek Bhute is currently pursuing the final year of the Diploma in Information Technology at Pimpri Chinchwad Polytechnic, Pune. His technical interests include full-stack enterprise systems, database indexing topologies, and algorithmic fairness auditing.

ABOUT THE AUTHORS

1. Ms. Poonam Ternikar is a Lecturer and Project Guide in the Department of Information Technology at Pimpri Chinchwad Polytechnic, Pune, Maharashtra, India. Her academic and research specializations include Software Engineering, Advanced Web Technologies, Database Systems, and Cloud Architectures.
2. Atharva Teli is currently pursuing the final year of the Diploma in Information Technology at Pimpri Chinchwad Polytechnic, Pune. His research interests include Natural Language Processing, Abstract Syntax Tree document parsing, and modern full-stack web architectures.