# AI-DRIVEN AUTOMATION OF THE JOB APPLICATION PROCESS: THE JOBPILOT SYSTEM

Mahir Tahirović,
[0009-0002-8307-9713]

Ulfeta Marovac*
[0000-0001-7232-3755]

State University of Novi Pazar,
Novi Pazar, Serbia

**Abstract:**

The modern recruitment landscape is characterized by a growing asymmetry between job seekers and employers, largely driven by the widespread use of Applicant Tracking Systems (ATS). Candidates often submit 50–100 applications before receiving an interview opportunity, while a significant portion of applications is filtered out before reaching human recruiters. This paper presents JobPilot, an end-to-end automated system based on an AI vs AI approach, which leverages artificial intelligence to optimize job applications for automated screening systems. The system integrates job data acquisition, requirement analysis, and AI-based generation of personalized cover letters within a unified workflow. A key component of the system is an AI-driven process that performs semantic alignment between candidate profiles and job requirements, enabling context-aware personalization of application content. The evaluation was conducted over a three-week period involving seven users who actively used the system for job search and application submission. The results indicate a significant reduction in application time, with an average of 8.2 minutes per application, compared to traditional approaches that require substantially more time. Users reported improved efficiency and successful interview invitations, confirming the practical value of the system. These findings suggest that AI-driven automation can improve the efficiency of the job application process and support candidates in highly automated recruitment environments.

**Keywords:**

Artificial Intelligence, Job Application Automation, Large Language Models, Applicant Tracking Systems, Workflow Automation.

## INTRODUCTION

The digital transformation of recruitment has significantly reshaped the job application process, but has also introduced new challenges for job seekers. While platforms such as LinkedIn, Indeed, and Glassdoor provide access to a vast number of job opportunities, the process of applying remains time-consuming and inefficient. According to recent studies, candidates typically need to submit between 50 and 100 applications to receive a single interview opportunity [1]. Considering that preparing a personalized application requires between 30 and 60 minutes [2], this results in a substantial investment of time, often exceeding 50 hours per interview.

Correspondence:
Ulfeta Marovac

e-mail:
umarovac@np.ac.rs



DOI: 10.15308/Sinteza-2026-632-637

A major contributing factor to this inefficiency is the widespread use of Applicant Tracking Systems (ATS), which automatically filter and rank applications based on predefined criteria. These systems rely on keyword matching and machine learning models, leading to the rejection of a large proportion of applications before they reach human recruiters. It is estimated that approximately 75% of applications are filtered out due to insufficient alignment with job requirements [3].

In such an environment, where artificial intelligence is used to evaluate candidates, it is reasonable to apply similar technologies to improve the quality and effectiveness of job applications. This paper presents *JobPilot*, an automated system based on the AI vs AI paradigm, which leverages large language models

(LLMs) to generate semantically optimized and personalized application content tailored to specific job descriptions. JobPilot addresses several key challenges in the job application process, including time inefficiency, low response rates, lack of scalable personalization, limited tracking of submitted applications, and the psychological burden associated with repetitive tasks. The system automates the complete workflow, from job data acquisition and requirement analysis to the generation and submission of personalized applications, allowing candidates to focus on interview preparation and skill development.

The main objective of this work is the design and implementation of a functional prototype that enables end-to-end automation of the application process. The system integrates AI-based personalization, modular and scalable architecture, and mechanisms for tracking and evaluating application effectiveness.

The contributions of this paper are threefold. First, it introduces an AI-driven approach for optimizing job applications in ATS-dominated environments. Second, it proposes a unified workflow that integrates data acquisition, analysis, and content generation. Third, it provides an empirical evaluation of the system based on real-world usage.

The remainder of this paper is organized as follows: Section 2 provides a comparative analysis of existing solutions, Section 3 describes the system architecture and main components of JobPilot, Section 4 presents the evaluation and results, and Section 5 concludes the paper.

## 2. COMPARATIVE ANALYSIS AND IDENTIFIED LIMITATIONS OF EXISTING SOLUTIONS

Current solutions for job seekers can be broadly categorized into four main groups: professional networks (e.g., LinkedIn [2]), job aggregators (e.g., Indeed [4]), specialized AI tools (e.g., LazyApply [5]), and application optimization platforms (e.g., JobScan [3]). While these systems provide partial support for the job application process, they differ significantly in terms of automation, personalization, and integration capabilities, often addressing only specific stages of the workflow. Professional networks such as LinkedIn [2], with over 900 million users, provide extensive access to job opportunities and features like “Easy Apply,” which facilitate high-volume submissions but lack sufficient personalization for more specialized roles. Job aggregators such as Indeed [4] offer a wide range of listings collected from multiple sources, increasing visibility but still requiring manual effort during the application process. Platforms such as Glassdoor [6] contribute valuable insights into company culture and salaries, yet they do not support automation of the application process. Specialized AI tools, including Sonara [7] and LazyApply [5], have emerged to address the need for increasing application volume. While these tools enable automated or semi-automated submissions, they often rely on generic templates and prioritize quantity over quality, limiting their effectiveness in competitive hiring environments. In contrast, application optimization platforms such as JobScan focus on improving resume alignment with job descriptions, but they lack integration with automated application workflows. Overall, existing solutions, provide only partial support for the job application process, differing significantly in terms of automation, personalization, and system integration, and typically addressing only specific stages of the workflow. Table 1 presents a comparison between existing solutions such as LinkedIn, LazyApply, and JobScan and the proposed JobPilot system.

As shown in Table 1, existing solutions exhibit significant limitations. Platforms such as LinkedIn provide access to job listings but lack automation and personalization. Tools like LazyApply enable large-scale application submission but rely on generic templates, while JobScan focuses on optimization without supporting automation.

These limitations indicate a trade-off between scalability and personalization. The proposed system, JobPilot, addresses this gap by integrating all stages of the application process into a unified pipeline.

**Table 1.** Comparative analysis of existing solutions and the proposed JobPilot system

| Functionality | LinkedIn Premium | LazyApply | JobScan | JobPilot |
|---|---|---|---|---|
| Automatic Job Search | No | Yes | No | Yes |
| AI Personalization | No | Generic | No | Yes |
| Application Automation | No | Yes | No | Yes |
| Multiple Job Sources | LinkedIn Only | Limited | No | Expandable |
| Application Tracking | Basic | No | No | Yes |
| ATS Optimization | No | No | Manual | Automatic |

## 3. SYSTEM ARCHITECTURE AND MAIN COMPONENTS

JobPilot is built as a distributed system based on a modular, microservice-oriented architecture designed to support scalable and efficient automation of the job application process. The system integrates a web-based frontend, backend services, persistent data storage, workflow orchestration, external data acquisition, and AI-based content generation.

The frontend layer provides a user interface for defining job search criteria, managing user profiles, and tracking submitted applications. It enables real-time interaction with the system and communicates with backend services through REST APIs. The backend layer acts as the central coordination component. It processes user requests, applies business logic, and triggers workflow execution through the orchestration layer. It also manages authentication and application tracking, ensuring consistent system behavior across all components.

The data layer stores user profiles, job searches, collected job postings, and application data. Persistent storage is supported by a relational database, while caching mechanisms improve performance and responsiveness during system execution.

The system is implemented using modern web technologies. The frontend is developed using React and TypeScript, while the backend is built on the Spring Boot framework, providing robust support for RESTful services. Data persistence is handled through a PostgreSQL relational database, and Redis is used for caching and session management in order to improve system performance.

The process of job data acquisition is based on automated collection of job postings from external platforms such as LinkedIn using external scraping services (e.g., Apify [8]). The collected data are cleaned, normalized, and transformed into a structured format suitable for further processing. This process is illustrated in Figure 1.

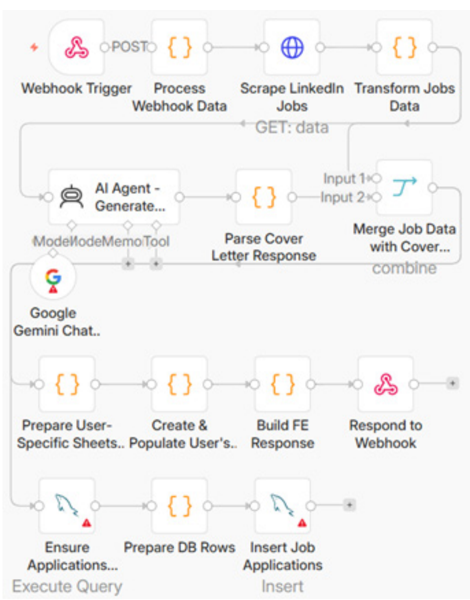


**Figure 1.** Data acquisition process from external job platforms

The core functionality of the system is implemented through a workflow orchestration layer based on the N8N platform [9], which coordinates the application pipeline. This layer integrates data acquisition, processing, AI-based content generation, and application submission into a unified and automated workflow. The process is illustrated in Figure 2, which represents the automated job application submission workflow.

To ensure system reliability, basic fault-handling mechanisms are implemented, including retry strategies for data collection failures and fallback mechanisms in cases of unavailable AI services.

The AI module represents the core innovation of the system and is responsible for generating personalized cover letters based on job descriptions and candidate profiles. The content generation process includes the extraction of job requirements, semantic matching with candidate skills, and final text generation.

The generation process is guided by structured prompts that combine job requirements, candidate experience, and contextual information about the target company. For example, a simplified prompt can be defined as follows:

```
“Generate a professional cover letter
for a [job title] position based on the
following requirements: [job require-
ments]. The candidate has experience in
[skills] and [years of experience]. The
tone should be formal and tailored to
the company [company name].”
```

This process is illustrated in Figure 3, which presents the AI-based content generation workflow and the interaction between system components and the language model.

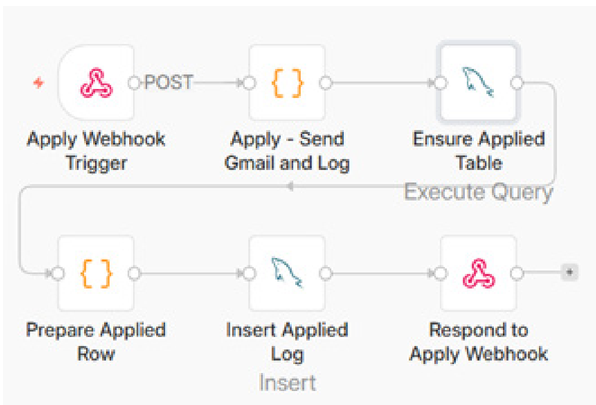


**Figure 2.** Automated workflow for job application submission in JobPilot

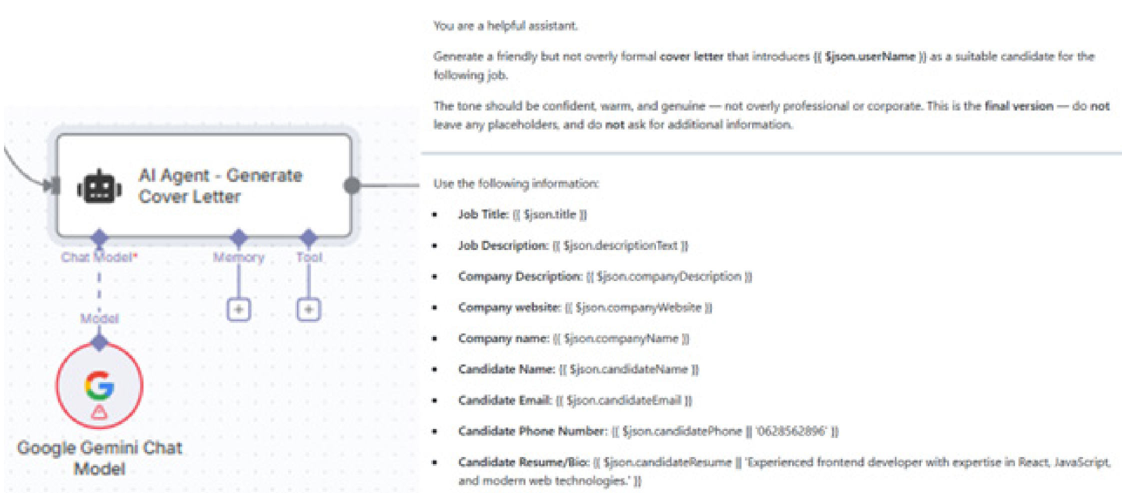


**Figure 3.** AI-based content generation process

The language model employed in the system is Gemini 2.5 Pro [10], integrated via the Google API. A significant architectural characteristic of the AI module is that the LLM integration is structurally isolated from the remaining system components, enabling concurrent processing of multiple requests. Consequently, the system supports the simultaneous execution of multiple model instances processing independent requests. This design facilitates horizontal scalability and allows for model substitution without introducing changes to other parts of the system.

To support quality assurance and systematic performance monitoring, each submitted application is recorded as an audit log entry containing operationally relevant metadata. Personally identifiable information (PII), including name, surname, email address, and phone number, is explicitly excluded from these records. The audit log enables both manual and automated review of generated content, supporting the tracking of relevant quality metrics such as output accuracy and degree of personalization.

Finally, the system includes an application delivery component responsible for sending generated applications and updating their status, completing the end-to-end application cycle. Together, these components form an integrated platform that automates the entire job application process, from data acquisition to AI-based content generation and application submission. The selected figures illustrate the key stages of the system, including data collection (Figure 1), automated application submission (Figure 2), and AI-based content generation (Figure 3).

## 4. EVALUATION AND RESULTS

The evaluation and validation of the JobPilot system were conducted through a combination of functional testing and a real user experiment, with the aim of assessing system reliability, efficiency of the application process, and the quality of generated content. The testing was carried out over a three-week period with a group of seven users who actively used the system for job search and application submission. System validation was performed across multiple testing levels, including unit testing, integration testing, and end-to-end scenarios. Unit testing confirmed the correctness of key functional components, particularly in data processing and validation logic. Integration testing verified stable communication between backend services, the database, and the N8N workflow engine, without data loss or execution interruptions. End-to-end testing demonstrated that the complete process—from job search to application submission—can be executed continuously without critical errors. These results confirm that the system operates reliably and consistently under realistic usage conditions.

Within the user experiment, participants performed a total of 100 job searches and submitted 72 applications, resulting in three interview invitations. The average time required to prepare and submit a single application was 8.2 minutes, which represents a substantial improvement compared to the traditional process that typically takes between 35 and 45 minutes. A detailed overview of user activity and performance metrics is presented in Table 2. These results indicate that the system significantly improves efficiency while maintaining content quality and relevance.

The quality of AI-generated cover letters was evaluated through user feedback and manual content analysis.

**Table 2.** Results of the user experiment

| User | Job Searches | Applications Sent | Interviews | Avg. Time (min) |
|---|---|---|---|---|
| K1 | 12 | 8 | 0 | 7.2 |
| K2 | 18 | 14 | 1 | 8.1 |
| K3 | 9 | 6 | 0 | 9.4 |
| K4 | 15 | 11 | 1 | 7.8 |
| K5 | 21 | 16 | 0 | 8.5 |
| K6 | 11 | 7 | 1 | 7.9 |
| K7 | 14 | 10 | 0 | 8.3 |
| **Total** | **100** | **72** | **3** | **8.2** |

The average quality rating was 4.4 out of 5, with users highlighting the relevance of the generated content and its alignment with specific job requirements. In addition, users reported a significant time saving, averaging approximately 18.5 hours over the testing period, which further demonstrates the practical value of the system in real-world job search scenarios. Overall, the evaluation results confirm that JobPilot effectively automates the job application process, significantly reduces the time required for application preparation, and maintains a high level of content quality. This confirms the practical applicability of the proposed approach in real-world recruitment scenarios.

## 5. CONCLUSION

This paper presented JobPilot, a system for automating and optimizing the job application process using workflow orchestration and AI-based content generation. The system addresses key challenges such as time inefficiency and lack of personalization in modern recruitment. Evaluation results show that JobPilot significantly improves efficiency, reducing the average application time to 8.2 minutes while maintaining high content quality (4.4/5). The system also proved to be reliable through multi-level testing and effective in real-world usage, where users achieved interview invitations and reported substantial time savings. These findings confirm that combining automation with large language models can enhance the job application process and improve candidate competitiveness. JobPilot demonstrates that end-to-end automation of the job application process is technically viable and provides measurable advantages. Future work will focus on improving system scalability, ensuring compliance through integration with official job platforms, and reducing operational costs through the use of alternative language models. Additionally, future developments include enhanced evaluation mechanisms and extended system capabilities, such as multilingual support and broader platform integration.

## REFERENCES

[1] Zippia Research, "Job Search Statistics 2023," 2023. [Online]. Available: https://www.zippia.com/advice/resume-statistics/

[2] LinkedIn Talent Solutions, "The Time Cost of Job Applications," 2023. [Online]. Available: https://www.linkedin.com/pulse/quality-quantity-your-job-applications-james-kingham/

[3] JobScan, "ATS Statistics: How Many Resumes Get Past ATS?," 2024. [Online]. Available: https://www.jobscan.co/applicant-tracking-systems

[4] Indeed, "About Indeed," 2024. [Online]. Available: https://www.indeed.com/about

[5] LazyApply, "LazyApply Official Website," 2024. [Online]. Available: https://lazyapply.com/

[6] Glassdoor, "About Glassdoor," 2024. [Online]. Available: https://www.glassdoor.com/about

[7] Brian, "Sonara AI Review: Pros, Cons & Best Alternative," Jan. 31, 2025. [Online]. Available: https://jobcopilot.com/sonara-best-alternative/. [Accessed: Nov. 10, 2025].

[8] Apify, "Apify SDK and Scraping Documentation," 2024. [Online]. Available: https://docs.apify.com

[9] N8N GmbH, "N8N Documentation," 2024. [Online]. Available: https://docs.n8n.io

[10] Google, "Gemini API documentation," 2024. [Online]. Available: https://ai.google.dev/gemini-api/docs. [Accessed: Nov. 10, 2025].