



Thapovan
Info Systems Inc.
Intense Pursuit of Knowledge

**Your Dedicated IT Solution Partner in
US ✦ India ✦ Europe ✦ Middle-East**

**AI POWERED
PAIR PROGRAMMING**

Case Study



New Jersey - Chennai - Palakkad



+91-9150049548



bd@thapovan-inc.com



www.thapovan-inc.com



Problem Statement

Onboarding a new resource into a project has its dependencies such as providing a walkthrough of the system, both from the business perspective & the code, educating them about the standards that are followed for the project or the organization and the dependency of senior or experts in the project.

Challenges

- **Time-Consuming Manual Processes:** Developers had to manually navigate JIRA for task tracking and GitLab for code management, which slowed down productivity.
- **Onboarding New Developers (Knowledge Transfer):** When new developers joined the project, they required detailed KT sessions from senior developers. This process took valuable time and was sometimes insufficient for the juniors to understand from the knowledge sharing.
- **Doubt Clarification:** Junior developers often needed to consult with seniors on specific issues, which added to the workload of senior team members and delayed task resolution.

Solution

By integrating an AI-powered project assistant with a LLM, we enhanced both our task management and developer collaboration, making onboarding smoother and improving the efficiency of doubt resolution through JIRA and GitLab integration.

Onboarding & Knowledge Transfer:

- When a new developer joined the project, they could query the system for documentation or past decisions, e.g., “What’s the design approach for the login feature?” or “Explain the database structure.”
- The project assistant provides clear, structured responses based on the project documentation. Which reduces the dependency on senior developers for basic KT, making the onboarding process faster and more consistent.



Doubt Clarifications:

- Junior developers often had specific doubts, such as “What approach should I take for handling errors in the API module?” or “How was this bug fixed in the last sprint?”
- The assistant could pull relevant past decisions, code examples, or JIRA tickets related to similar issues, saving senior developers from repeatedly answering the same questions. If the assistant could not resolve the issue, it escalated the query to the senior developers, tagging them in the appropriate JIRA or GitLab context.

Benefits



- **Faster Onboarding:** New developers could self-serve a significant portion of the onboarding process by querying the assistant for documentation and KT materials.
- **Reduced Dependency:** The LLM-powered project assistant could resolve a significant number of common queries and doubts from junior/new developers, allowing senior developers to focus on high-priority tasks.
- **Enhanced Knowledge Management:** The system acted as a repository of past decisions, code examples, and project documentation.
- **Improved Developer Productivity:** The assistant made it quick and easy to access relevant tickets, branches, or commits with natural language queries.

Technology Environment:



Thapovan was responsible for identifying and implementing the most appropriate technology for the various applications in order to meet the performance, cost and usability objectives set by the client.

Major Technology Components:

Language: Python (3.x), JavaScript

Models: Open AI 4o mini, Open AI 4o, Llama 3.1, and Open AI 4 (Based on requirement)



Webserver: Nginx

Database: Postgres

Caching Technology: Redis

Frameworks: Fast API, Langchain, Next JS

Conclusion



By integrating project assistant for JIRA and GitLab, we addressed not only the inefficiencies in task management and code review but also streamlined onboarding and improved communication between junior and senior developers. The project assistant provides a smoother KT experience, quicker doubt clarification, and empowered developers to work more efficiently.