FUJITSU

Takara Agentic AI

Accelerating the Software Development Lifecycle

FUJITSU-PUBLIC

Challenge

Development teams are being asked to do more and more with large, diverse code bases. Developers face difficulty speeding the Software Development Lifecycle without wasting time writing repetitive code or tests, all the while keeping consistent code quality and adhering to standards. They would like to leverage AI. However, they are concerned about keeping their code base secure and out of the public cloud.

Solution

The Takara Agentic AI Developer Platform is a powerful tool designed to securely accelerate the Software Development Lifecycle by boosting developer productivity and improving code quality while keeping data and code secure within your infrastructure. From intelligent code suggestions and advanced code analysis to an interactive Q&A system, Takara Agentic AI offers comprehensive support for developers at all stages of the Software Development Lifecycle.

Takara Agentic AI brings together intelligent code generation, documentation analysis, UI creation and AI assisted question and answers all within your IDE (Integrated Development Environment) or through a web browser.

Code Intelligence Agent:
Explain, Optimize, Clean, Test, Secure Code
Low Level Design generation from Code, Integration to Mermaid and PDF.

Interactive AI Question and Answer:
Multi turn conversations, history tracking and context aware answers.

UI Generator Agent:
Convert Screenshots /Prompts or Figma diagrams to UI Code (HTML, CSS)
Upload a screenshot or just describe it in English.

Design Agent (HLD & LLD Generation):
Generate Sequence Diagrams and Mermaid Charts

RAG (Retrieval Augmented Generation):
RAG server-side fine tuning for accurate domain specific answers.

Document Review Agent:
Validate Documentation, Ensure Consistency

Code Debug Agent:
Real-Time Autosuggestions

Takara Agentic AI is not just an AI tool; it is your bridge to smarter development.

A Day in the Life of a Developer and Takara Agentic AI

Janet is a developer who has received a request to implement a new feature in an existing enterprise application. The codebase is large, includes legacy components, and spans multiple services. Before writing a single line of code, Janet uses **Takara's** knowledge of the code base to analyzes the relevant code modules.

Takara Agentic AI:
- Highlights high-complexity areas
- Flags potential integration risks
- Surfaces related legacy patterns used elsewhere in the system

Takara Agentic AI gives developers immediate insight into scope, impact, and risk.

Planning & Estimation
Janet leverages Takara Agentic AI to generate an **AI Generated Implementation Plan and Action Items** that includes a to do list and estimated efforts for the action items.
This reduces planning time and increases delivery confidence.

UI Generation
The UI Designer provides a Figma design for the new feature. Janet uses **Takara's UI Generation** capability to automatically generate UI code directly from the Figma project, aligned to the application's existing frameworks and standards.

Test-Driven Development
Before implementing the feature logic, Janet uses Takara Agentic AI to generate unit tests and identify key test scenarios and edge cases. Takara Agentic AI enables **Test Driven Development (TDD)** from the outset improving overall code reliability.

Intelligent Code Completion
As Janet codes up the new feature, Takara Agentic AI with its **Intelligent Code Completion** acts as a **real-time AI peer programmer**. Janet prompts Takara Agentic AI to generate code that is aligned with the design patterns of the existing codebase. Additionally, Takara Agentic AI recommends naming conventions and error-handling approaches.
At all times developers remain fully in control, accepting or ignoring Takara's suggestions as needed.

Debugging
When parts of the code aren't working as expected, using **Takara Code Intelligence** Janet has Takara Agentic AI explain the existing logic and identify why the strange behaviors are occurring. Takara Agentic AI accelerates root-cause analysis without breaking developer flow.

AI Code Review
Before finalizing the feature, Janet does a Code Review with Takara Agentic AI. As part of the review Janet specifically has Takara Agentic AI **scan for potential security vulnerabilities**. Janet fixes a concern that Takara Agentic AI highlights.

Janet then uses Takara's **Intelligent Documentation Review** to update the documentation and validate that it is consistent with the code changes.

Code Check-in
Finally, Janet uses Takara Agentic AI to generate the Code Repo Check-in Message based on her team's message template. Janet checks-in the code and their CI/CD pipeline deploys the new feature.

Customer use case

Customer
A large Canadian provincial government agency

Situation
The customer was interested in how they could leverage Takara Agentic AI to enhance their application development and legacy modernization. This is a challenging, large public-sector environment with complex legacy applications and requiring strict compliance. They wanted to use AI coding tools but needed strict security and couldn't risk their code being sent out to the cloud.

Solution
Fujitsu deployed the Takara Agentic AI Developer Platform on the customer's infrastructure to drive AI-assisted development workflows and support legacy modernization. They leveraged Takara's configurable agents for code analysis, refactoring, and documentation.

Impact

30–70% reduction in manual development and support effort. Faster and compliant modernization of legacy systems, with full data control on customer-controlled infrastructure.