# FeatureInsight

## Overview

GenAI-powered assistant that analyzes new features, user stories, or JIRA tickets to uncover potential impacts, risks, and updates.

## Problem Statement

In Agile environments, teams struggle with understanding the ripple effects of new features across layers (UI, API, DB, etc.). Effort estimation, sprint planning, and testing readiness often suffer from a lack of contextual visibility.

## Objectives and Scope

Goals:

* - Understand impacted modules, services, and data flows.
* - Improve estimation and grooming confidence.
* - Identify testing and documentation needs.

Scope includes developer inputs such as user stories or JIRAs.

## Target Users

- Developers  
- Scrum Leads  
- Tech Leads

## Key Features

- Analyze code modules, DB schema, and business logic  
- File/function level impact estimation  
- Identify tech debt hotspots  
- Flag test and documentation gaps

## Benefits

1. Sprint Planning – Accurate estimation and resource alignment  
2. Impact Visibility – UI, API, DB, cloud layer awareness  
3. Faster Grooming – Save time scanning codebases  
4. Improved QA Readiness – Flag regression needs  
5. Documentation Awareness – Identify design artifacts needing update  
6. Scope Creep Detection – Spot risky ‘small tweaks’

## High-Level Architecture

JIRA Input → Embedding Engine → Codebase Analyzer → Output: FeatureInsight Report

## Technologies & GenAI Elements

- Python, LangChain, FAISS, LLMs (OpenAI or OSS)  
- IDE plug-in or Web UI  
- Vector-based similarity and context search

## Risks and Mitigation

- False positives/negatives → Mitigate via human feedback  
- Security of embeddings → Use masking + encryption  
- Adoption barrier → Integrate non-intrusively into existing tools

## Timeline and Milestones

Phase 1 (Month 1): Proof of Concept  
Phase 2 (Month 2): Alpha with embeddings + feedback UI  
Phase 3 (Month 3): Beta testing with 1–2 teams  
Phase 4 (Month 4+): Production with security + CI integrations