**Problem Statement**

While GitHub Copilot excels at code completion, it lacks the contextual awareness of the problem developers are solving.

Developers often need to understand project requirements and locate dependent assets—such as documentation, design files, or configuration data—before they can effectively use Copilot.

This creates friction in the development process, as they must leave the development environment to access critical resources which can lead to inefficiencies.

**Proposed Solution**

We propose integrating Smart Agents with GitHub Copilot to create a seamless development experience.

* Access requirements directly from JIRA using **Atlassian Rovo.**
* Build a dedicated **smart agent for Discover called Volt** that can search, retrieve, and manage Discover assets allowing developers to find everything they need effortlessly.
* Combine the contextual capabilities of **smart agents with Copilot’s code completion**, enabling more accurate and informed code suggestions.

**User Experience – Context Diagram**

A diagram of a company

Description automatically generated

**Volt Service Architecture**

A screenshot of a computer

Description automatically generated

* The Volt Agent Service is built using langchain and langgraph module.
* It consists of Supervisor and Worker agents. The supervisor agent handles conversation with user and worker nodes.
* The worker agents retrieves Discover assets such as from Alation,Sharepoint on internal vector DB.
* The agent that uses ReAct prompting. Based on paper “ReAct: Synergizing Reasoning and Acting in Language Models” (https://arxiv.org/abs/2210.03629)
* Volt has an authentication service to authenticate user the first time.

Setup info can be found at: <https://github.com/mayur11235/volt_service>