

Question 1:

Given that Byterover aims to help developers capture and reuse the context they need while programming, what are the possible intentions or use cases developers might have when using Byterover?

Additionally, what would an ideal integration of Byterover into the development workflow look like to minimize friction and make it feel natural within the developer's existing tools and processes?

Question 2:

Given that Byterover is currently implemented as a Command Line Interface (CLI), what technical solutions could enable the Byterover agent to communicate seamlessly with target coding agents such as Cursor or Claude Code?

How can this integration be designed to minimize friction for developers, ensuring that Byterover feels like a natural extension of their existing coding environment rather than an additional tool to manage?

Question 3:

Assume your proposed technical direction in Question 2 has been approved. Please proceed to implement a prototype that meets the following requirements:

1. **Context Capture Agent**

Implement an agent that captures contextual information from the user and stores it in a **context tree**. You can refer to our blog for more details about the context tree concept.

<https://www.byterover.dev/blog>

2. **Command Line Interface (CLI)**

Develop a CLI that allows users to run the agent directly from the terminal. The CLI should support connecting the agent to a target coding assistant such as Claude Code or Cursor.

3. **Implementation and Submission**

The implementation must be written in TypeScript. Once completed, please share your solution with us via a GitHub repository.