No documentation?
No direction?
No problem.

Nodify

Code visualization, made simple

Computer Science

McMaster University

What is Nodify?

Nodify is a VSCode extension that generates high-level execution flow diagrams for any Python or TypeScript codebase. With Nodify, developers can instantly visualize interactions between different components with zero hassle. This living, interactable form of documentation grows alongside the software development process and makes use of the latest LLMs to provide intelligent summaries where applicable. Through Nodify's autonomous diagram generation, engineers can see the progression of a codebase in real-time, catching design flaws long before they cause problems in published code.

Background ›

Goals ›

Background

Professional software engineering teams often make use of hand-drawn diagrams and flowcharts to document the high-level architecture of a large project. Concrete visual aids of this nature are especially useful at reducing the mental strain and onboarding time for new developers, accelerating codebase proficiency and increasing overall productivity within the team. However, such flowcharts must be created and updated manually, quickly becoming dated and unmaintainable as agile project requirements inevitably change. Therefore, Nodify seeks to automate this diagram-creation process to better serve developers in debugging, documenting, and understanding their projects.

Goals

As a productivity-focused tool, the goals of Nodify are as follows:

- Complete automation of the documentation process for execution flow diagrams.
- Interactable diagrams that can expand and contract based on level of desired detail.
- Zero interruptions and non-invasive implementation to the normal development workflow.

How Does Nodify Work?

The Nodify pipeline is a straightforward process from source file to graph, witch caching to prevent wasteful LLM calls during the summarization process:

1. Open project in VSCode
2. Run the Nodify command
3. Parse source code into a Nodify-focused AST
4. Partition code into separate blocks
5. Generate summaries for each block
6. Generate the Nodify Webview graph

Why Nodify?

Nodify can instantly generate interactive up-to-date flowchart documentation from any Python or TypeScript codebase, virtually eliminating the drawbacks of creating visual aids for communicating programs. Contributors and users alike will be able to instantly grasp the high-level functionality of any program. Additionally, in approaching this problem with a non-invasive stance, our project can run anywhere, anytime, even on incomplete code. Nodify can even provide retroactive documentation for undocumented projects, generating flow charts to easily conceptualize different components of a program. As a result, users of Nodify should expect much less frustration and wasted time when working with foreign or familiar projects.

Results and Limitations ›

Future Work ›

Nodify's Efficacy

On small-to-medium sized projects using GPT-4o, Nodify provided intuitive, clear graphs of an entire codebase within a few seconds. Further, collapsible nodes and a collision-avoidance pathing algorithm within the graph allowed for reasonable simplification and organization of complex diagrams respectively. As part of Nodify's goal for a zero-interruption workflow, we experimented with a custom fine-tuned model to run Nodify on local GPUs. Hardware quality, specifically GPU VRAM, was one of the largest limiting factors for local LLMs, preventing the use of higher quality models or requiring prohibitively longer running times on larger codebases.

Overall, weaker generic models tended to create more code partitions with slightly inaccurate summarizations, reducing the effectiveness of Nodify as a high-level analysis tool. Likewise, even after mechanically decomposing blocks of code into smaller chunks and using the best models available (Claude 3, GPT-4o, etc.), some very large-scale Python projects struggled to translate well into Nodify diagrams, requiring significant amounts of processing time or providing oversimplified summaries of the project.

Where to Find Nodify?

Nodify is available for free on the VSCode marketplace! Download it from the extensions tab and you're all set.

Note: Several Nodify features require an OpenAI API key or a self-hosted local LLM with Ollama. Complete guides for both methods are available in the README.

What's Next for Nodify?

As LLMs become better at summarization and semantic textual analysis, Nodify's diagram output will also continue to provide higher quality information. Currently, Nodify mirrors programs by converting project source code into high-level flow graphs, but this concept can be feasibly reversed. That is, instead of writing code and visualizing the resulting architecture, AI may be able to generate programs from high-level models of a software project. Tools like GitHub Copilot and Cursor already exist to generate code to a certain degree, but often require detailed prompts, comprehensive context, and low-level requirements analysis to prevent hallucination. Instead, Nodify users will be able to prototype a project fully in the graphical node editor, with AI agents applying design changes directly to a codebase when requested.

Tooling

python TS AST-GREP

Brought to You By...

- Youcef Boumary (@youcefs21)
- Ahmet Dumlu (@ahmetdumlu)
- Mutaz Helal (@MutazHelal)
- Anthony Hunt (@Ant13731)