During a programming change task, developers often need to understand where a certain variable or object is referenced and how it is used in order to implement the change.

For languages TypeScript and JavaScript the only available tools to find these points of interest are tools to find variable and property references and to display a call tree. These tools, however, start losing viability in larger codebases, where they can return hundreds or even thousands of results sorted only by the source file names.

In this thesis, we have developed an extension for Visual Studio Code that finds these references and categorise them into groups based on usage. It can also perform similar analysis on call arguments based on the position of the argument. We evaluated our solution and showcased it in several real-world use cases.