



IDE-IT (Backend)

Eric Wahlquist, John Barcellos, David Thien

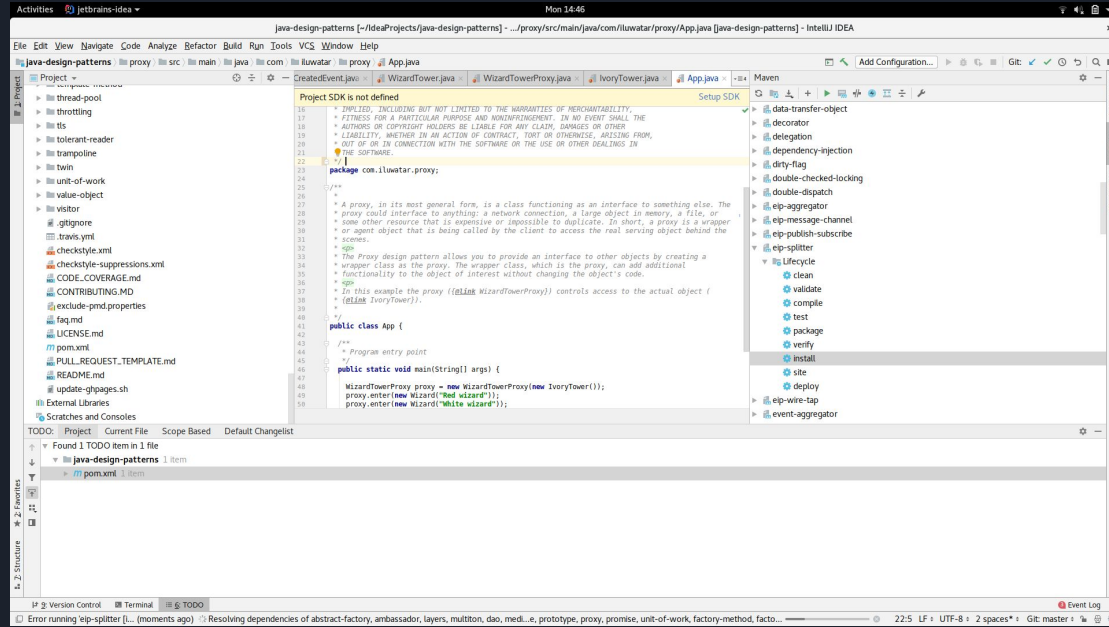
Motivation

IDEs are great...

- Automate common tasks
- Easy project navigation

...however

- Cumbersome
- Many features
 - Hard to find these features
 - Hard to even know these features exist





Problem

Discoverability of IDE features is poor

How it works now

- Daily tips
- Top 20 best features list
- Long tutorial online tutorials (slideshows, videos)
- Colleague sees you coding

Difficult to get relevant information at the relevant time



IDE-IT

Integrated Development Environment - Intelligent Tutorials

Eclipse plug in

- Tracks user input and provides suggestions for using IDE features

IDE-IT Back end

- Responsible for detecting user input and evaluating which IDE features the user might be interested in



Related Work

MouseFeed

- Informs users of hotkey shortcuts for IDE features
- Only tracks user mouse input in menus/toolbars

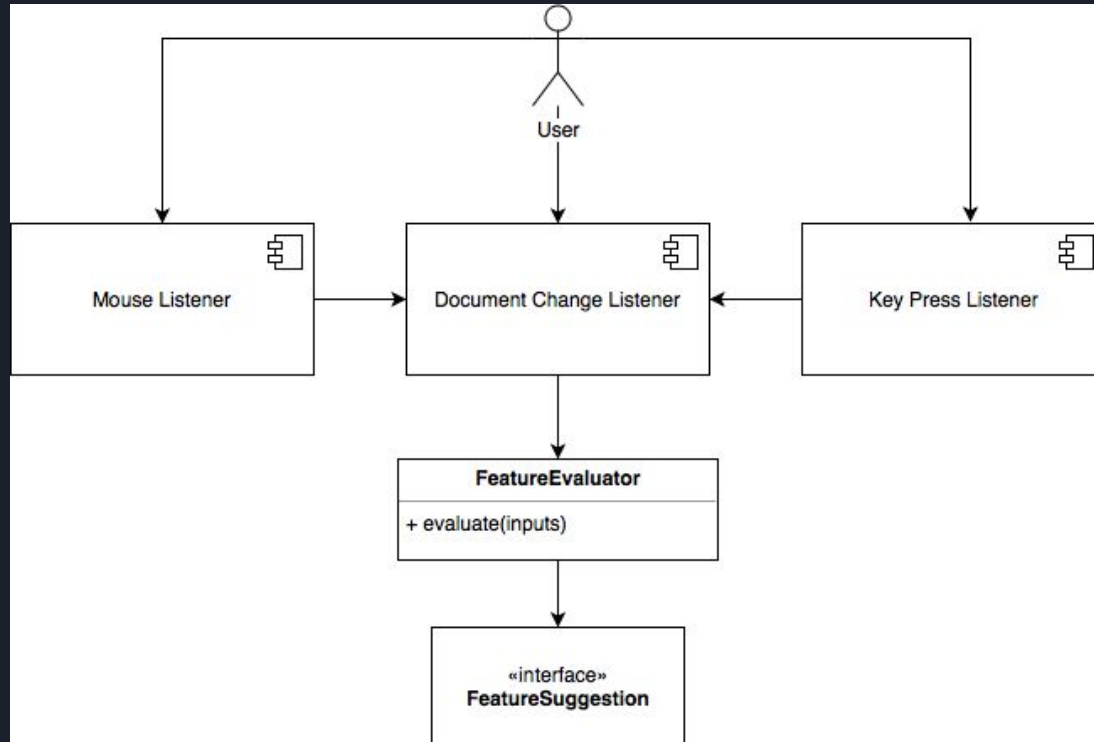
Intellij Features Trainer

- Helpful, but you still need to seek it out
- Interactive tutorials instead of just reading / watching a video
- Doesn't actively track user input to provide relevant info at relevant times

Intellij Idea Feature Suggester

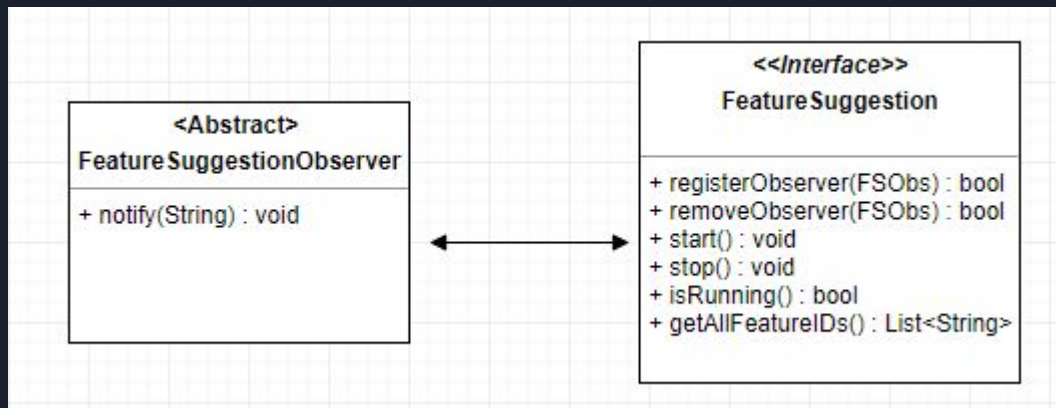
- Similar idea, but buggy and doesn't work very well
- No interface for a front end or a UI to display / control notifications

IDE-IT Backend Approach



IDE-IT Backend Interface

Observer Design Pattern





IDE-IT Backend Approach

How does this approach differ?

- Actively tracks user action within the default Eclipse document editor
- Provides a simple interface for an arbitrary front end
- Currently no Eclipse plugin provides this functionality



Evaluation

Our focus:

- Minimize false negative triggers
 - Have suggestions trigger when they would be helpful to a user
- Keep false positives low, but less of a concern

Evaluate through user testing:

- Ask users to act as if they don't know a feature exists
 - See what they do and if our plugin detects it properly



Preliminary results

- Implemented basic plugin framework
- Implement plugin dependencies
- Implemented basic document change and key press listeners



Difficulties/Challenges

- Learning curve / ramp up time for Eclipse plugin framework
- Considerable number of Eclipse APIs available
 - Functionalities are not immediately clear
- Fine tuning evaluation functions may be challenging
 - Each evaluation function is coded to look for a specific sequence of actions



Questions?