"Vim the IDE"
about.txt

— US Guy in Hamburg
— Software Engineer @ Yelp in Hamburg

disclaimer: opinions expressed are solely my own and do not express the views or opinions of my employer.
"Vim the Editor"
The IDE has many parts

IDE

Vim
Code Completion
Class Browser
Version Control
Debugging
Running Tests
Problems with the IDE

— too many features
— much complexity
— same with "All in 1 Vim Builds"
IDE Goals

Navigation

Editing

External Interactions
Navigation

Finding things
- in the current file
- other files
- searching via regex, symbols, or simple terms

You clone a new project and either
- know where you are going
- want to find someplace to go
Finding Things

— ctrlp https://github.com/ctrlpvim/ctrlp.vim

— py-matcher to increase speed https://github.com/FelikZ/ctrlp-py-matcher

The main fuzzy (and regex compatible) way to jump between files. Nearly instant with 60K files (linux source).
- go jump to file map gj
- buffer switching map gb
- recent files map gr
Project Navigation

— NERDTree https://github.com/scrooloose/nerdtree
— NERDTree Tabs https://github.com/jistr/vim-nerdtree-tabs
— vim-vinegar https://github.com/dhruvasagar/vim-vinegar

Always quickly available with '-' key. Easy to switch to files in the same directory or see where you are.
This file was moved to Documentation/admin-guide/README.rst

Please notice that there are several guides for kernel development.
These guides can be rendered in a number of formats, like HTML.

In order to build the documentation, use ``make html`` or
``make pdf``.

There are various text files in the Documentation/ subdirectory,
several of them using the Restructured Text markup notation.
See Documentation/INDEX for a list of what is contained in
it.

Please read the Documentation/process/changes.rst file, as it
contains requirements for building and running the kernel, and informations
about the problems which may result by upgrading your kernel.
— easy to modify files inline
— files are easier to modify with a "GUI"
Searching for things

— vim-grepper https://github.com/mhinz/vim-grepper

:Grepper -tool rg -query "foo"
2 main ways to search for things:
- quickfix search
- side search, with +3 lines context

Easy visual search
nmap gv <plug>(GrepperOperator)
Jumping to symbols

vim-gutentags makes tags work transparently
Editing

Mostly normal VIM things
- syntax highlighting
- linting
- code completion
Tab Complete

— SuperTab configure Tab complete
— vim-snipmate makes snippets work
— vim-snippets default snippets for everything
Easier jumping to characters

— vim-clever-f

Find the next character: type fe
Easy revisions

— vim-gundo
Easy alignment

— vim-lion
Working with tag based languages

— vim-surround
External Interactions

— Building software (compiling, make)
— Version Control
— Running tests
— Debugging
Building Software

set makeprg=make
Testing!

— https://github.com/benmills/vimux

The future:

— https://github.com/janko-m/vim-test
Structure your tests to mirror your code!

python/package/foo.py
tests/package/foo_test.py
Version Control

- **git-gutter**
- **vim-fugitive**
Debugging

uh oh
NeoVim

The future?
references.txt

— http://cewing.github.io/training.codefellows/assignments/day01/sublime_as_ide.html
— my dotfiles and vimrc github.com/askedrelic/dotfiles
Discussion is welcome!

@askedrelic
www.asktherelic.com