Understanding a Codebase Part 2

CS 691 / SWE 699
Fall 2025

GEORGE MASON UNIVERSITY

Logistics

- No reading questions this week
- Readings on testing due for next week
- Due 11/6 at 4:30pm:
  - Lecture 9 reading questions
  - Reflection 4
  - Project Checkpoint

Today

- In-Class Activity
  - Two tasks understanding code in a real open source codebase
- HW (due in 1 week before class)
  - Reflection 4

In-Class Activity

- Goal today: run an **experiment** comparing understanding code w/ and w/out Cursor
  - Will get 2 tasks, one without and one with Cursor
  - Work on a real open source codebase
  - Work **individually** rather than in a group
  - Try to make as much progress as you can (but will be graded on your reflection, not your progress on the task)
- Will **only** have class time today to work on the two tasks
  - Will do reflection as HW afterwards
- Use Kaltura to record your screen for all of your programming work, upload video at end of class
- Will get handout describing your task
- Have extra informed consent forms if you need one

Schedule for today

- Project setup
- Task 1
  - if finish early, start break
- 10 min break
- Task 2

Project Setup

Clone the repository with

```
git clone https://github.com/weirane/qutebrowser-reflection4
git remote set-url origin https://github.com/qutebrowser/qutebrowser
```

After you have cloned the repository, use command `python3 scripts/mkvenv.py` to set up a development environment. A virtualenv will be created in `./.venv` and you can run the code with `.venv/bin/qutebrowser` (or `.\.venv\Scripts\qutebrowser.exe` on Windows)

You don't need to set up unit tests this time. Just verify that you can run the code.

Background

qutebrowser is a keyboard-driven, vim-like browser written in Python and Qt.

Configuring qutebrowser: the qutebrowser allows users to set configuration options in a python file. You can specify the config file using the `-C` option, like:
`./.venv/bin/qutebrowser -C ./config.py` or
`.\.venv\Scripts\qutebrowser.exe -C config.py`

Command line system: features of qutebrowser need to be invoked using the command line system, which can be triggered by hitting the : key (Shift+;). You can then enter the command after the : in the input bar. For example, to open google.com, you can use the `:open` command. Just type `:open https://google.com`, and then hit Enter.

Task 1

- Follow instructions on your handout
- Remember to use Kaltura to record all of your programming work
- **Only use an LLM if specified on your handout**
- Write your answers in a doc and submit when done with the task through Canvas (Lecture 9 - Task 1)

10 min break

Task 2

- Follow instructions on your handout
- Remember to use Kaltura to record all of your programming work
- **Will switch - if using LLM, will switch to not using an LLM; if not using LLM, will use LLM**
- Write your answers in a doc and submit when done with the task through Canvas (Lecture 9 - Task 1)