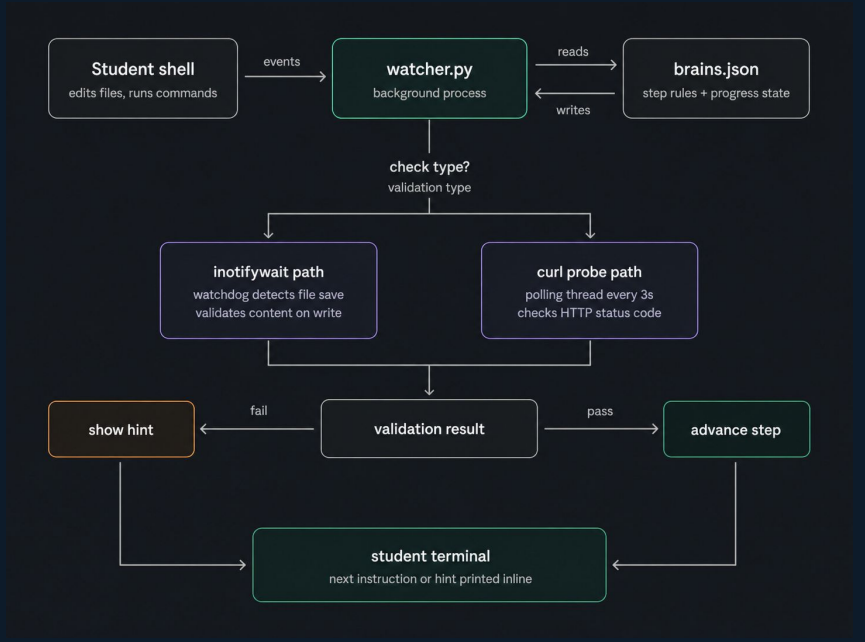
Sepehr R - Explore Minnie

Group 8 - Explore Minnie · Sepehr R · ID: 0647

watcher.py

Monitors student progress to deliver real-time CLI guidance

- Runs as a background process alongside the shell wrapper - detects file edits and HTTP responses to know exactly where the student is in the project, step by step
- Reads validation rules from brains.json for each step - inotifywait handles file changes, a polling thread checks HTTP endpoints every 3 seconds
- On validation: advances progress and prints the next instruction directly to the student's terminal.
- Also Contributed to: Bug fixes in tutor.sh, as well as coordinating with Thomas Mullaly in adding minnie to school network and setting up BIGGIE virtual machine



Code: github.com/B8ExploreM/Explore-Minnie/tree/watchdog - **Artifacts:** /home/sepehr/cs410/artifacts - **Deployed site:** https://sepehr-umb.github.io/minnie/

T-Shaped Skills: Evolution

Group 8 - Explore Minnie · Sepehr R · ID: 0647

BROAD - Ability to contribute across areas outside core specialty

UI/UX & Frontend

Functional interfaces and visual work - can contribute when needed

Team Leadership

Organized tasks, kept the bigger picture moving

DEEP

Git & Collaboration

Branching, merging, SSH keys, org repos - learned hands-on

Django + Apache

Deployed from scratch on Biggie VM

Scripting & Automation

Python scripting, threading, file I/O - watcher.py built this semester

Linux Server Admin

Account provisioning, permissions, sshd config, service management - applied directly on Biggie

nnnn: /home/sepehr/cs410/nnnn.txt

Additional Comments & Plans

Group 8 - Explore Minnie · Sepehr R · ID: 0647

On collaboration

Working in a team GitHub org, coordinating branches, and contributing without breaking others' work was the most practical Git experience I've had. We utilized a test brach called “canary” to consolidate all of our branches and bugfix before pushing to main

Next steps

Polish and bug fix watcher.py, run tests with the client, and work toward deploying the full tutor system to more users so that any student that has access to users can ssh into BIGGIE, have a user account created for them, and be able to practice the backend tutorial

On watcher.py

Architecture is solid and the passive validation layer works. Remaining work is polishing, bug fixing, and integration with the shell wrapper end-to-end.

Looking ahead

The interactive CLI tutor concept has real potential as a teaching tool. It's a great supplementation to the original minnie exercise. With further polish it could be a reusable framework for any backend development course.

Research paper: /home/sepehr/cs410/rpaper.pdf