

GitHub Portfolio Guide

A practical guide for presenting technical work clearly to recruiters, engineering managers, clients, and collaborators.

1. Choose What to Showcase

- [] Pin 4-6 repositories relevant to target work.
- [] Archive/unpin low-value experiments.
- [] Never expose confidential code.
- [] Use sanitized demo data when needed.

2. Repository README

- [] State the problem and intended users.
- [] Explain key features.
- [] List the stack.
- [] Include setup/run instructions.
- [] Add screenshots/demo links when useful.
- [] Document limitations.

3. Code Quality Signals

- ☐ Keep project structure understandable.
- ☐ Exclude secrets.
- ☐ Use environment variables appropriately.
- ☐ Keep naming consistent.
- ☐ Remove obvious debug artifacts.

4. Project Story

Problem / context

Technical decisions and trade-offs

Result / what I learned

5. Demonstrate Delivery

- [] Provide a working demo when practical.
- [] Include tests/quality checks where relevant.
- [] Document deployment approach.
- [] Explain API/data dependencies.

6. Data / Analytics Projects

- ☐ Describe data source.
- ☐ Document cleaning/transformation.
- ☐ Make visuals answer a clear question.
- ☐ Keep notebooks/scripts reproducible.

7. Frontend / Full-Stack Projects

- [] Demonstrate responsive behavior.
- [] Consider accessibility basics.
- [] Include loading/error states.
- [] Explain architecture/dependencies.

8. Profile-Level Polish

- [] Align bio to target roles.
- [] Make pinned projects tell a coherent story.
- [] Keep contact/LinkedIn/portfolio links current.
- [] Do not artificially inflate activity.

9. Portfolio Review Worksheet

Project	What It Proves	What to Improve

Next portfolio actions