Git:

A Getting Started presentation

by: Nap Ramirez
email: napramirez@gmail.com

What is Git?

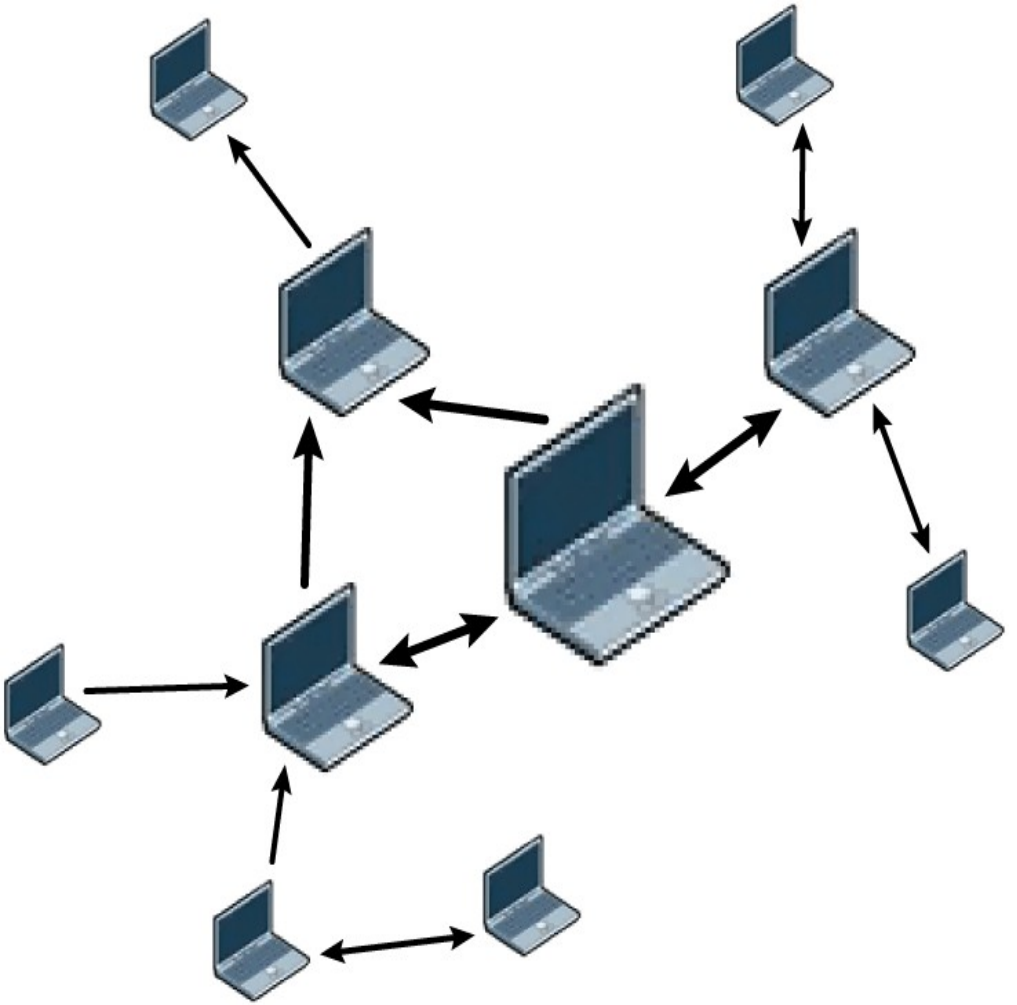
Git is a distributed version control system (DVCS).

Version control systems keep track of the changes in files and directories over time.

- records the history of data
- acts like a time machine of data

Architecture

Git is a distributed system, everyone has their own copy of the source codes.



GETTING USED TO...

Terminology

Repository

- a storage of the history of changes of the tracked files, hidden from the user

Working Copy

- the set of tracked files

Revision

- a specific state of data recorded in the history

GETTING USED TO...

Terminology (continued)

Branch

- a separate line of history of the tracked files deviating from the origin or another branch

Tag

- a reference to a specific state of a branch

LEARNING...

Basic Operations

Clone Remote Repository (git clone)

- acquire a copy of a repository
- `$ git clone git@git.mydomain.com:myproject.git`

Pull Updates From Repository (git pull)

- get updates from another repository and apply those into your repository
- `$ git pull origin master`

Pushing Changes To Repository (git push)

- send changes in your repository to another repository
- `$ git push remote master`

LEARNING...

Basic Operations (continued)

Create Branches (git branch <name>)

- creates a local branch based from the active branch
- `$ git branch feature1`

Switch Branch (git checkout <name>)

- switches active branch to any existing one
- `$ git checkout -b feature1`

Merging Branches (git merge)

- apply changes from specified branch
- `$ git merge feature1`

Perform/Apply Changes (git commit, git add)

- mark changes in history of active branch
- `$ git commit -am “changed the button label”`
- `$ git add README.txt`

LEARNING...

Feature-based Workflow

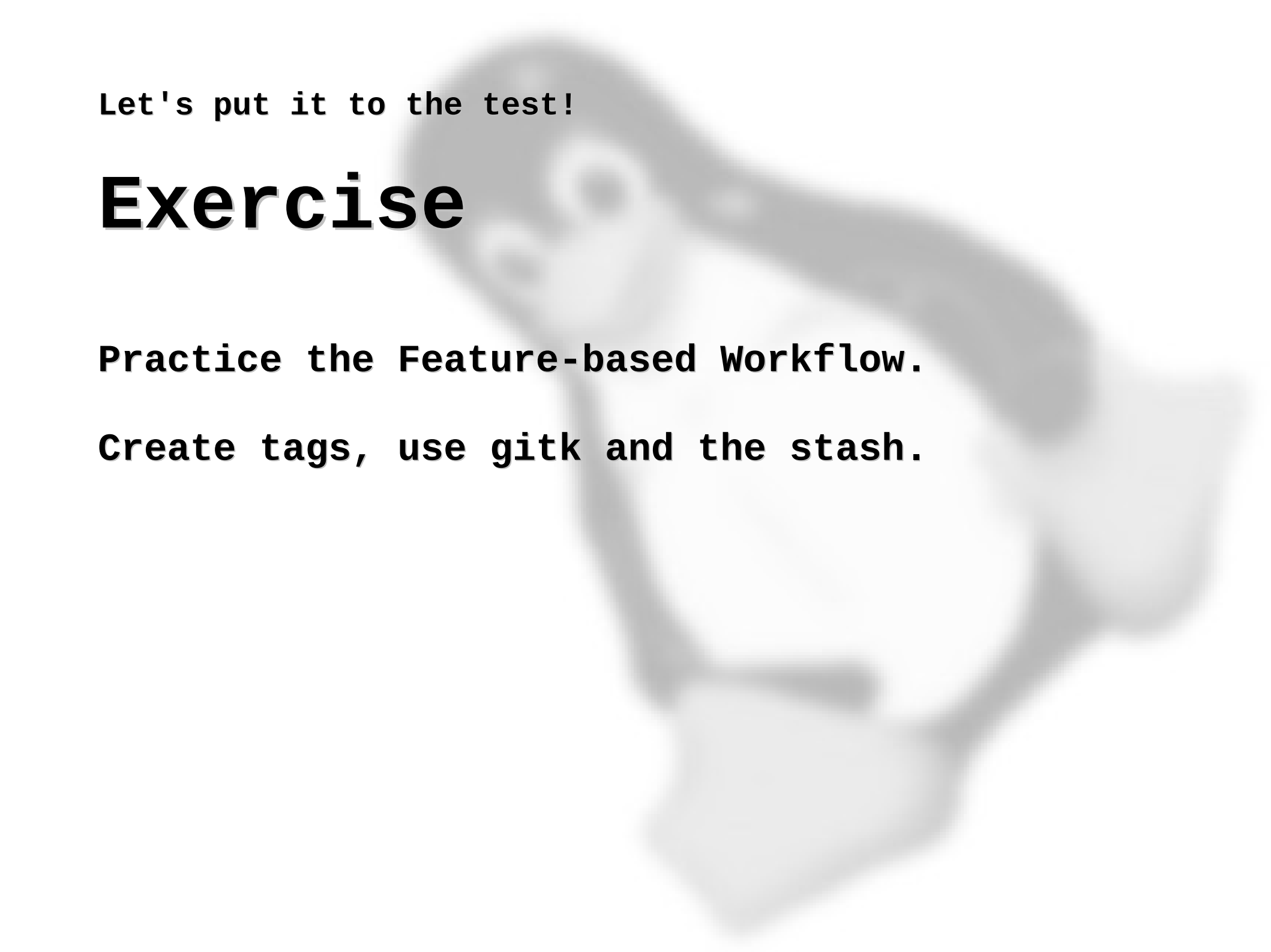
1. Pull from remote master into local master (in master, git pull)
2. Rebase local branch to changes from master (in local branch, git rebase master)
3. Perform changes, then commit to local branch (in local branch, git commit)
4. Merge to local master (in local master, git merge <local-branch-name>)
5. Push changes (in local master, git push)

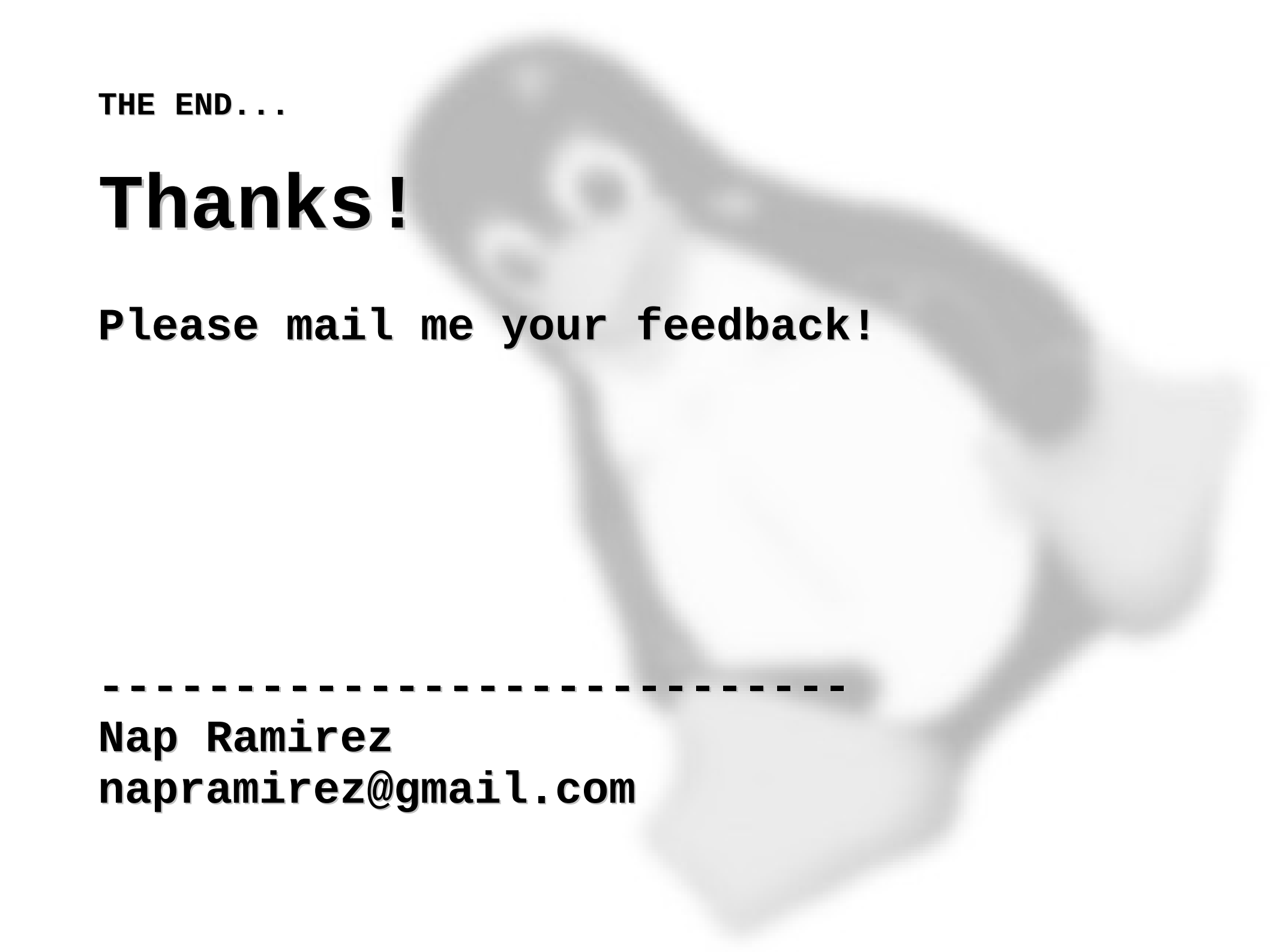
Let's put it to the test!

Exercise

Practice the Feature-based Workflow.

Create tags, use gitk and the stash.





THE END...

Thanks!

Please mail me your feedback!

Nap Ramirez
napramirez@gmail.com