Git, GitHub & SourceTree: The Backbone of Data-Driven Decisions

Essential tools for reproducibility, collaboration and workflow management in data science.

J by Jemael Nzihou

Git & GitHub – Basic Concepts

Git

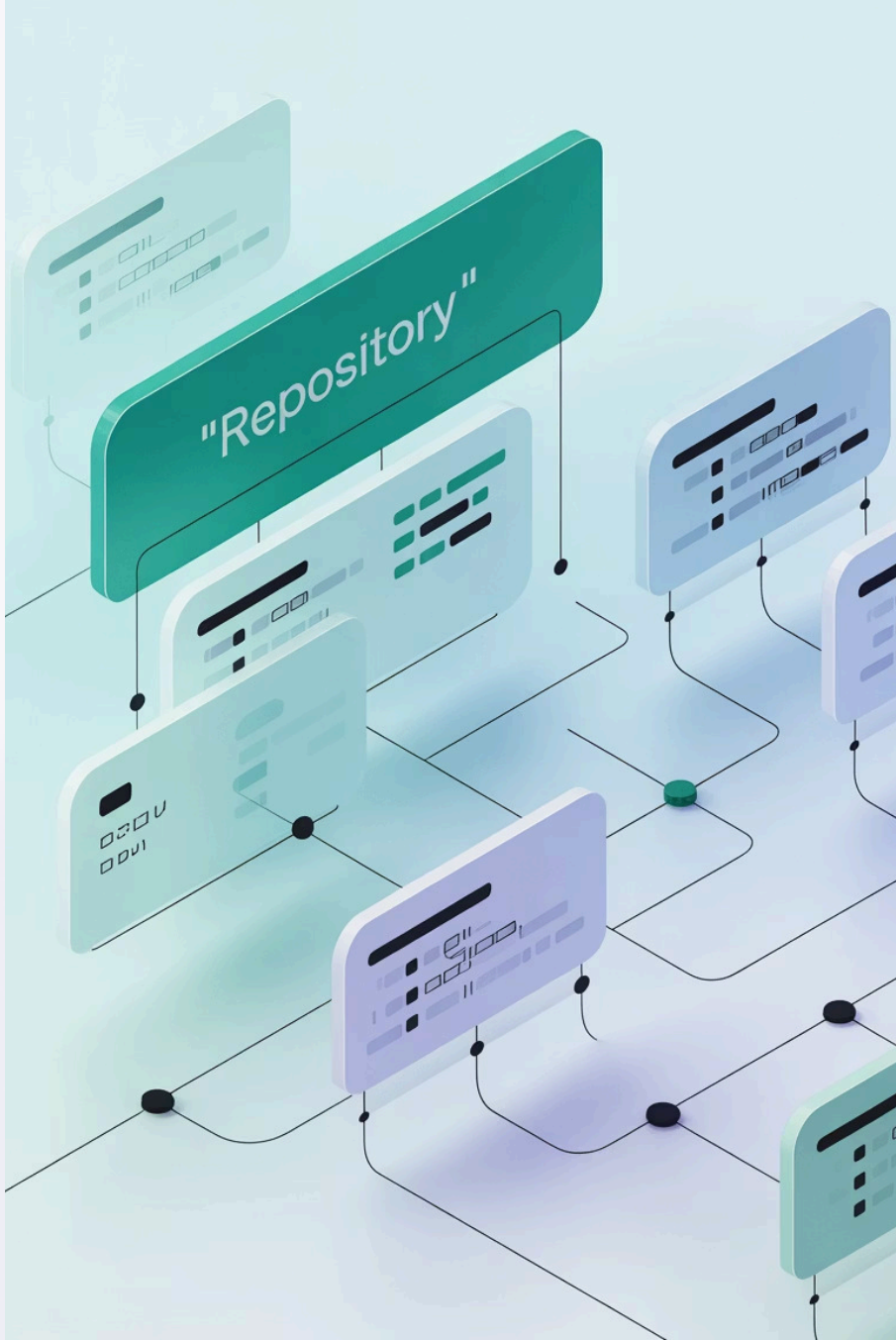
Version control system tracking file changes

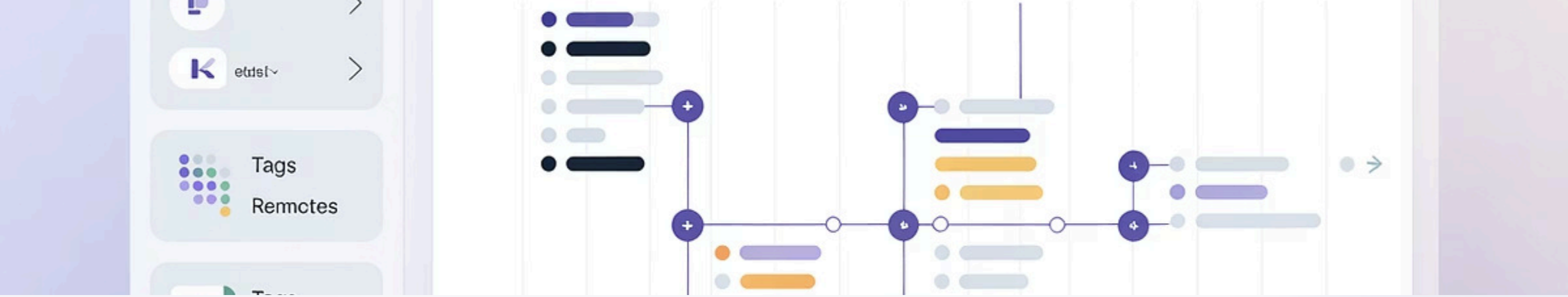
GitHub

Cloud-based host for Git repositories

Key Benefits

Collaboration, rollback, reproducibility





SourceTree Overview

GUI-Based Client

Visual interface for Git operations

Repository Connection

Links local repos with GitHub

Visual History

Clear commit and branch visualization

User-Friendly

Perfect for non-command-line users

Git Workflow Essentials

Common Workflows

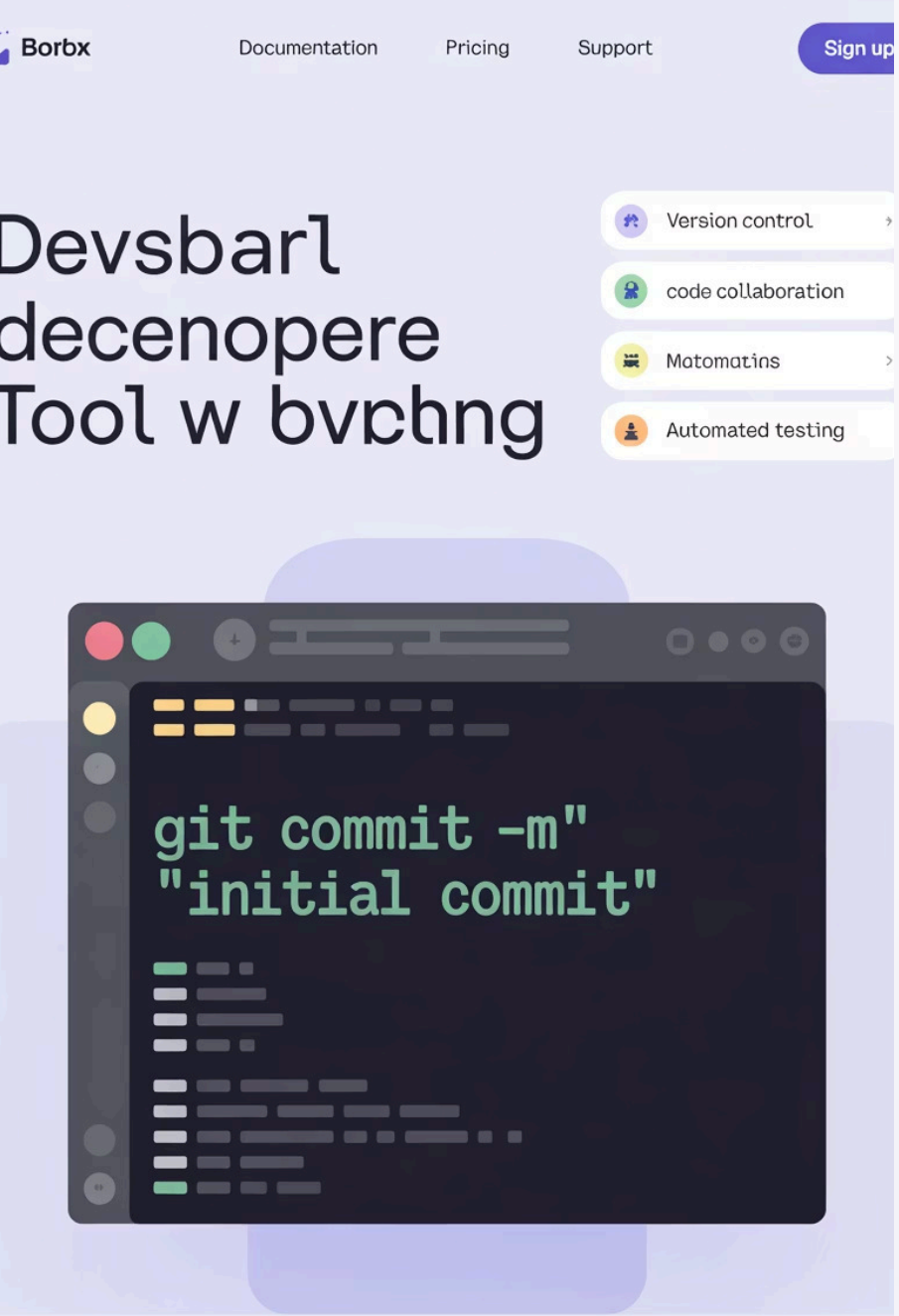
- Feature Branch
- Gitflow
- Forking

Core Components

- Branches
- Commits
- Pull Requests
- CI/CD

Purpose

- Code quality
- Minimal conflicts
- Team collaboration



Core Git Commands

Initialize

git init, git clone, git status

Stage & Commit

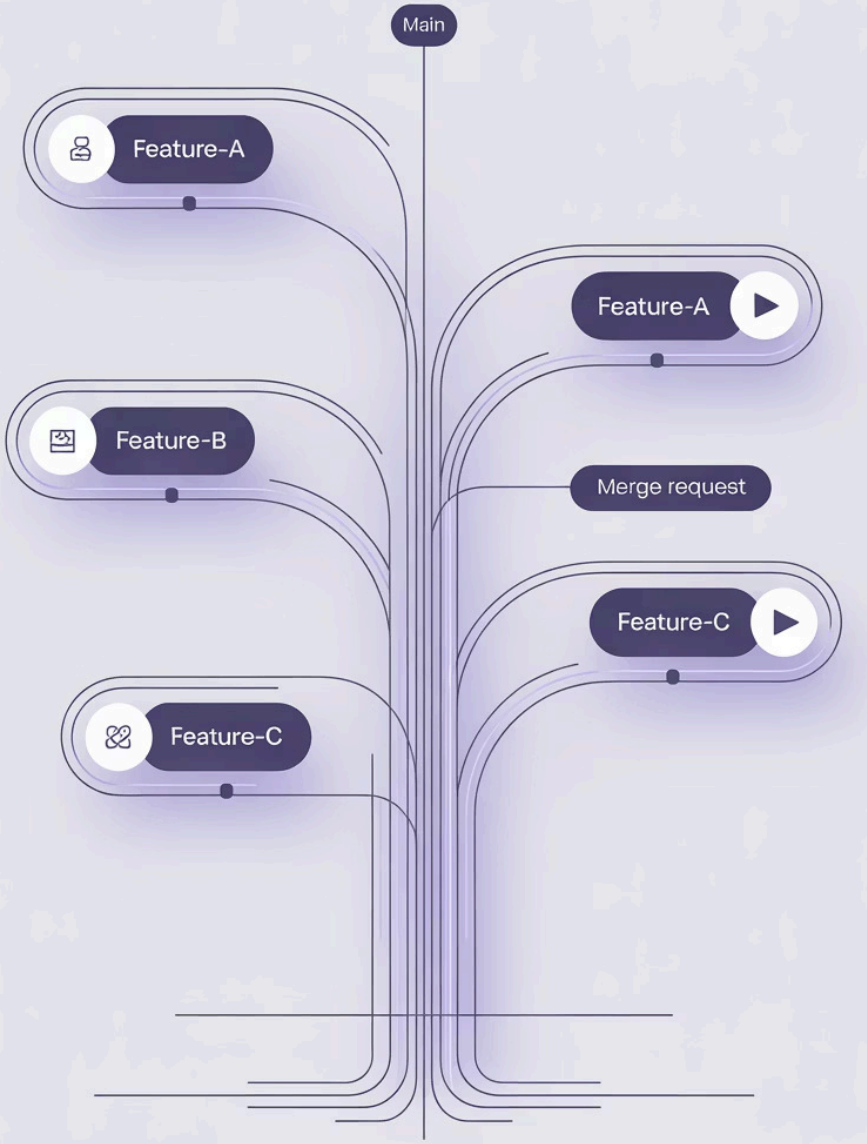
git add, git commit

Synchronize

git push, git pull, git merge

Full Control

Complete change management and sync



Git Branching Explained

1

Create Branch

Isolate feature development

2

Develop Safely

Work without affecting main code

3

Merge Changes

Integrate completed features

4

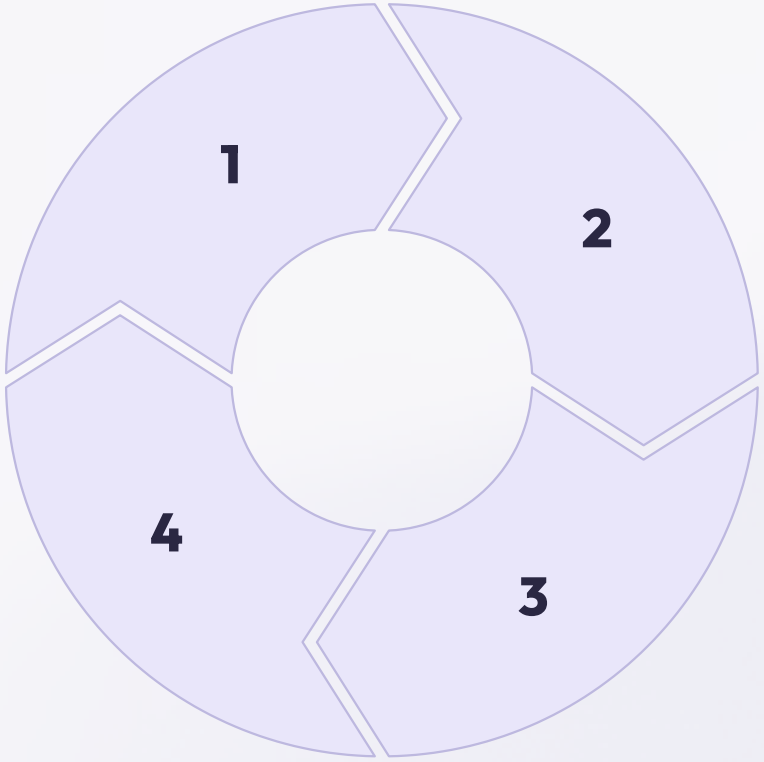
Delete Branch

Clean up completed work

Pull & Push Cycle

Pull

Fetch and merge from remote



Work Locally

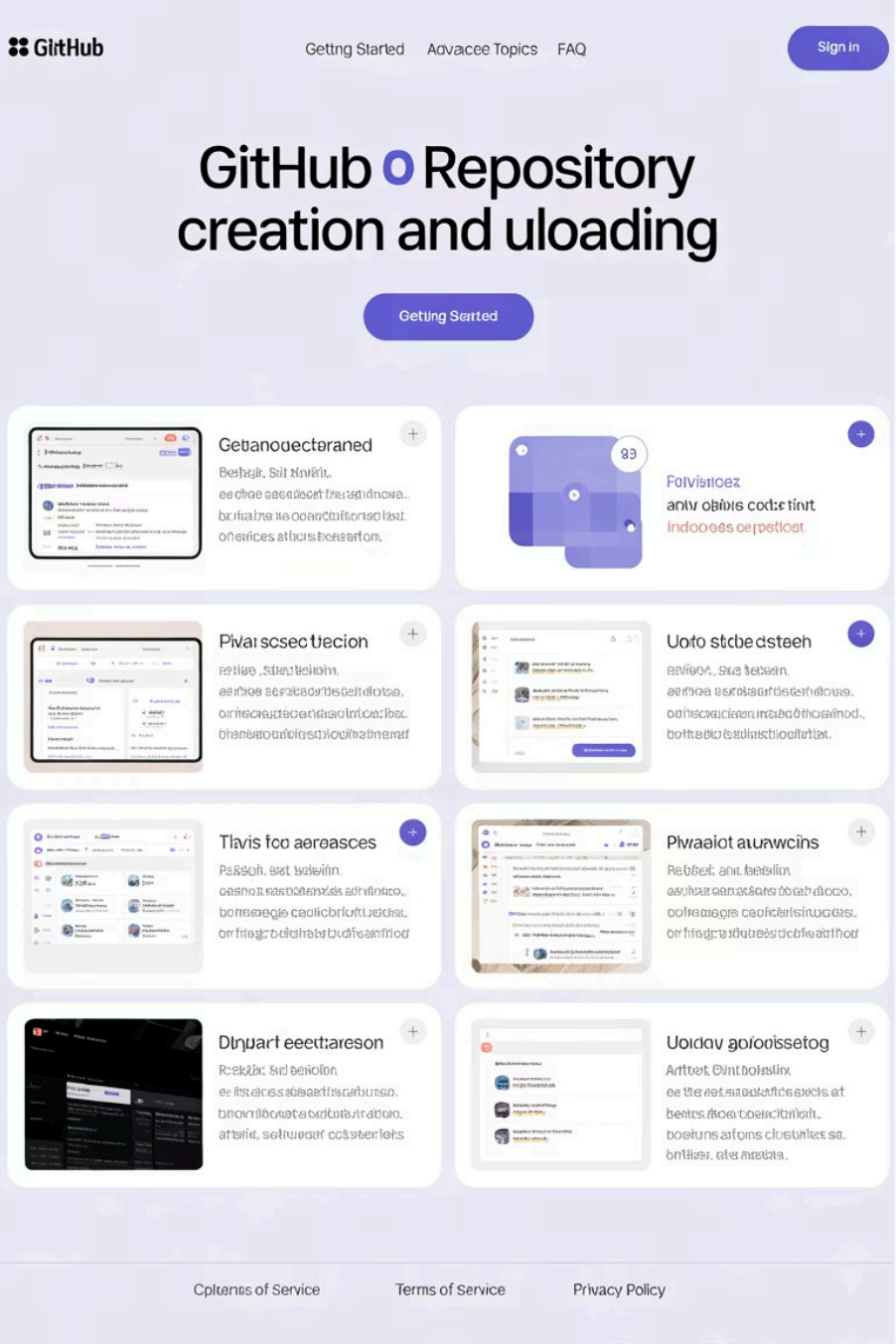
Make changes and commit

Push

Send commits to remote

Team Sync

Keep everyone updated



How to Push Files to GitHub

Initialize Repository

git init, git add ., git commit -m "initial commit"

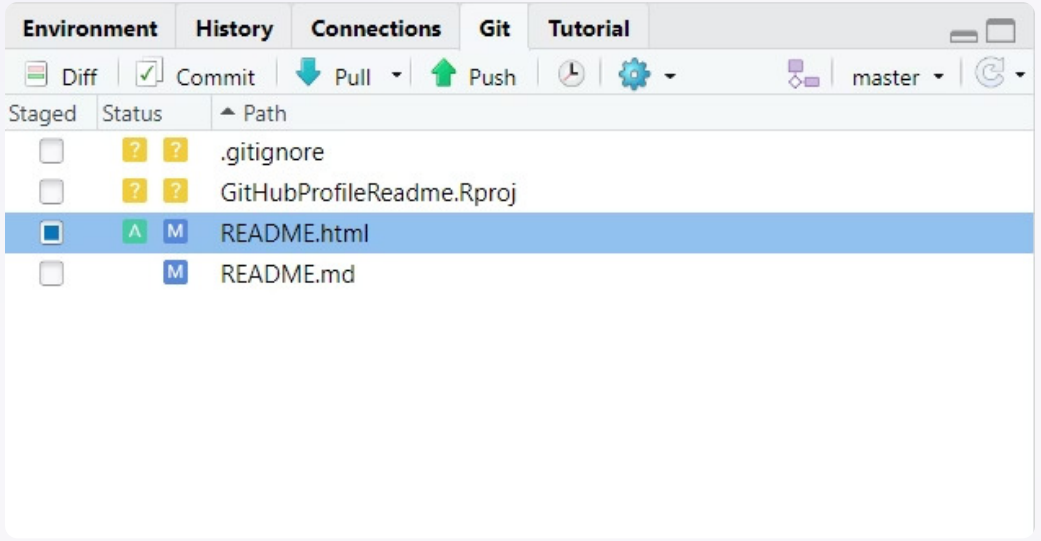
Connect Remote

git remote add origin https://github.com/user/repo.git

Push to GitHub

git push -u origin main

Perfect for uploading Jupyter notebooks, R scripts, SQL queries, and dashboards.



Git in R Programming

R developers can perform Git operations directly from R using packages like 'usethis' and 'git2r'.

- **Pull:** Use usethis::pr_pull() or git2r::pull() to fetch and merge remote changes
- **Commit:** Track changes with git2r::commit() after staging files
- **Push:** Send commits to GitHub with git2r::push() or through RStudio's Git panel

RStudio provides integrated Git tools for seamless version control in data science workflows.

README.md* | README_Predictive_Maintenance_GitH...

Preview on Save | Preview | Run

Source | Visual

```
<!-- badges: start -->
<!-- badges: end -->

# 👋 Hi there, I'm Jemael Nzihou

🎖️ Veteran | 🧪 Chemical Engineer | 📊 Data Scientist | ⏳ Lean Six Sigma Black Belt
Building bridges between **science, strategy, and social impact** – with a commitment to excellence and
transformation.

---

### 🔍 About Me

I'm a US Air Force veteran and a multidisciplinary professional with a background in:

- **Chemical Engineering (B.S.) & Chemistry**
- **Applied Business Analytics (M.S.)**
- **Data Science & BI Certifications (IBM, Google, Coursera)**
- **Lean Six Sigma Black Belt & Certified Quality Champion**

```

Git Software & Sectors

1

Development Tools

RStudio, JupyterLab, VSCode, PyCharm, GitHub Desktop

2

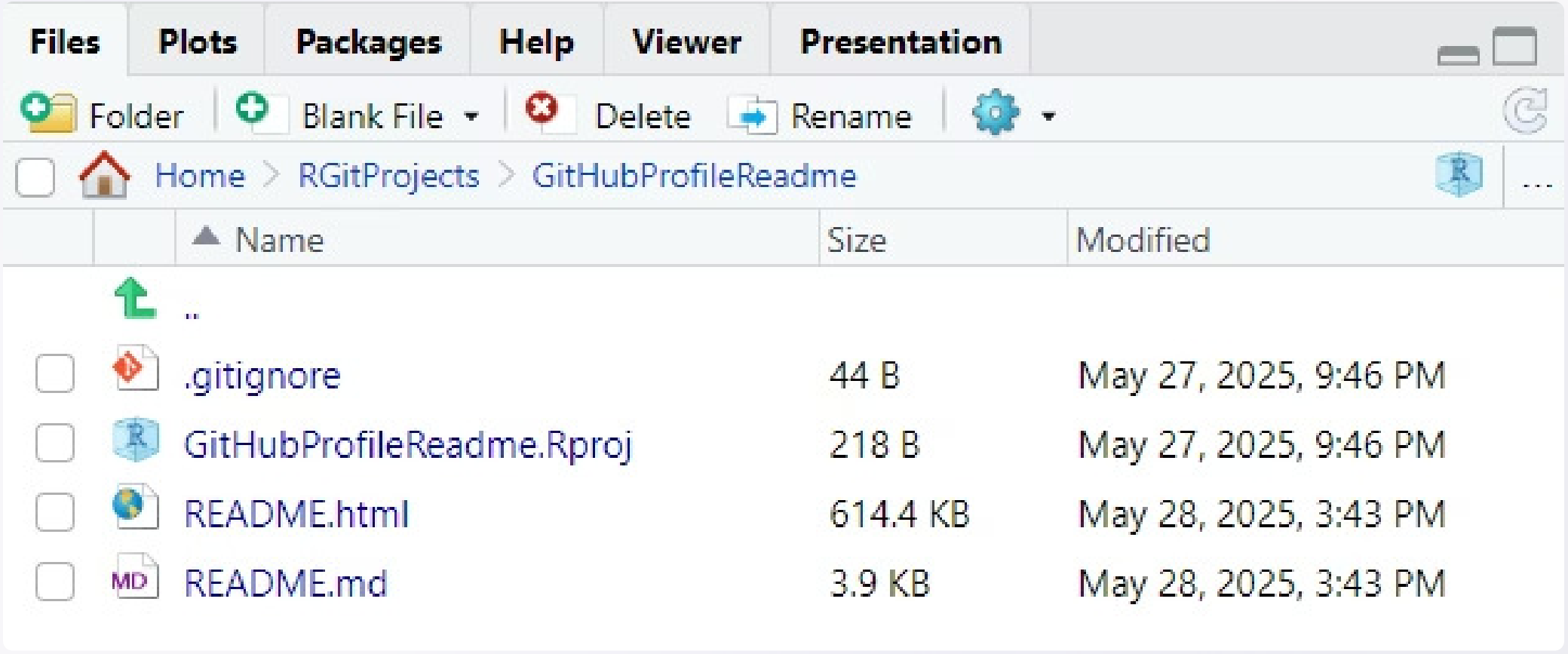
Industry Adoption

Healthcare, Finance, Manufacturing, Academia

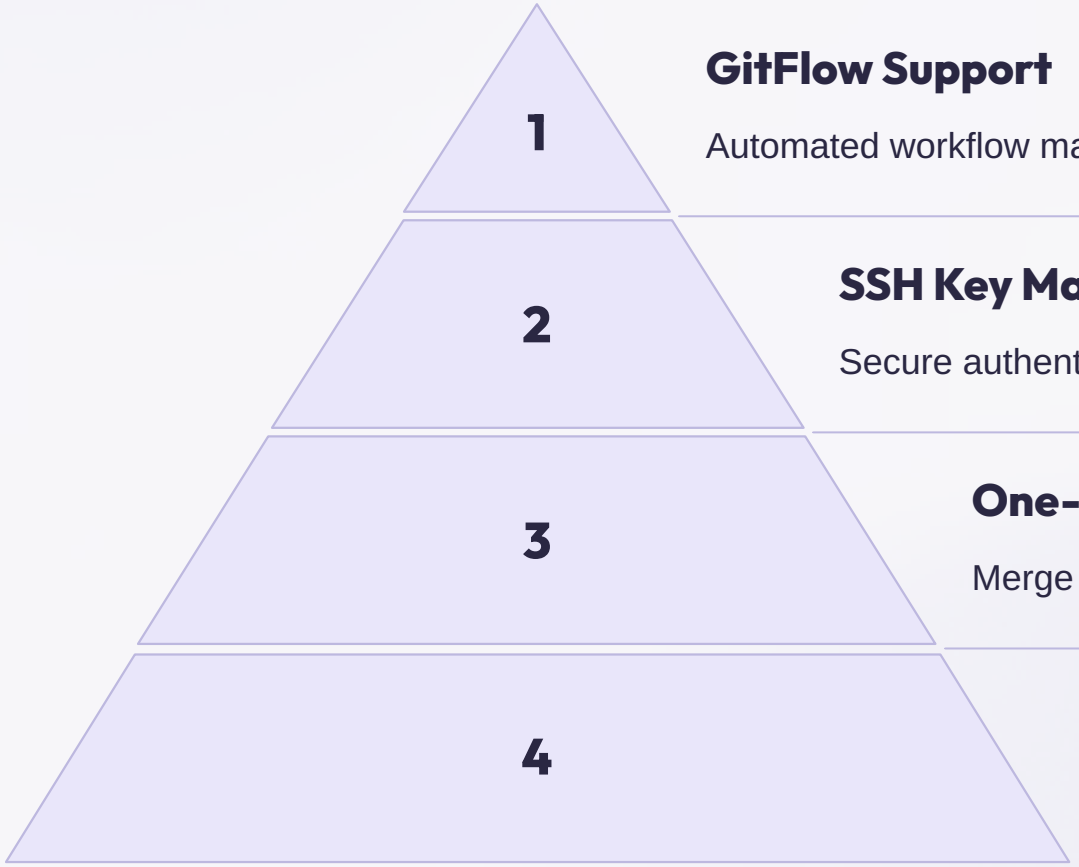
3

Critical Benefits

Auditability, reproducibility, data governance


Files
Plots
Packages
Help
Viewer
Presentation
Folder
Blank File
Delete
Rename
Home
RGitProjects
GitHubProfileReadme
Name
Size
Modified
..
.gitignore
44 B
May 27, 2025, 9:46 PM
GitHubProfileReadme.Rproj
218 B
May 27, 2025, 9:46 PM
README.html
614.4 KB
May 28, 2025, 3:43 PM
README.md
3.9 KB
May 28, 2025, 3:43 PM

SourceTree Features



GitFlow Support

Automated workflow management

SSH Key Management

Secure authentication handling

One-Click Operations

Merge and rebase simplified

Visual Branching

Clear repository visualization

Git in R Programming

1

Configure Git

Set up RStudio version control

2

Connect GitHub

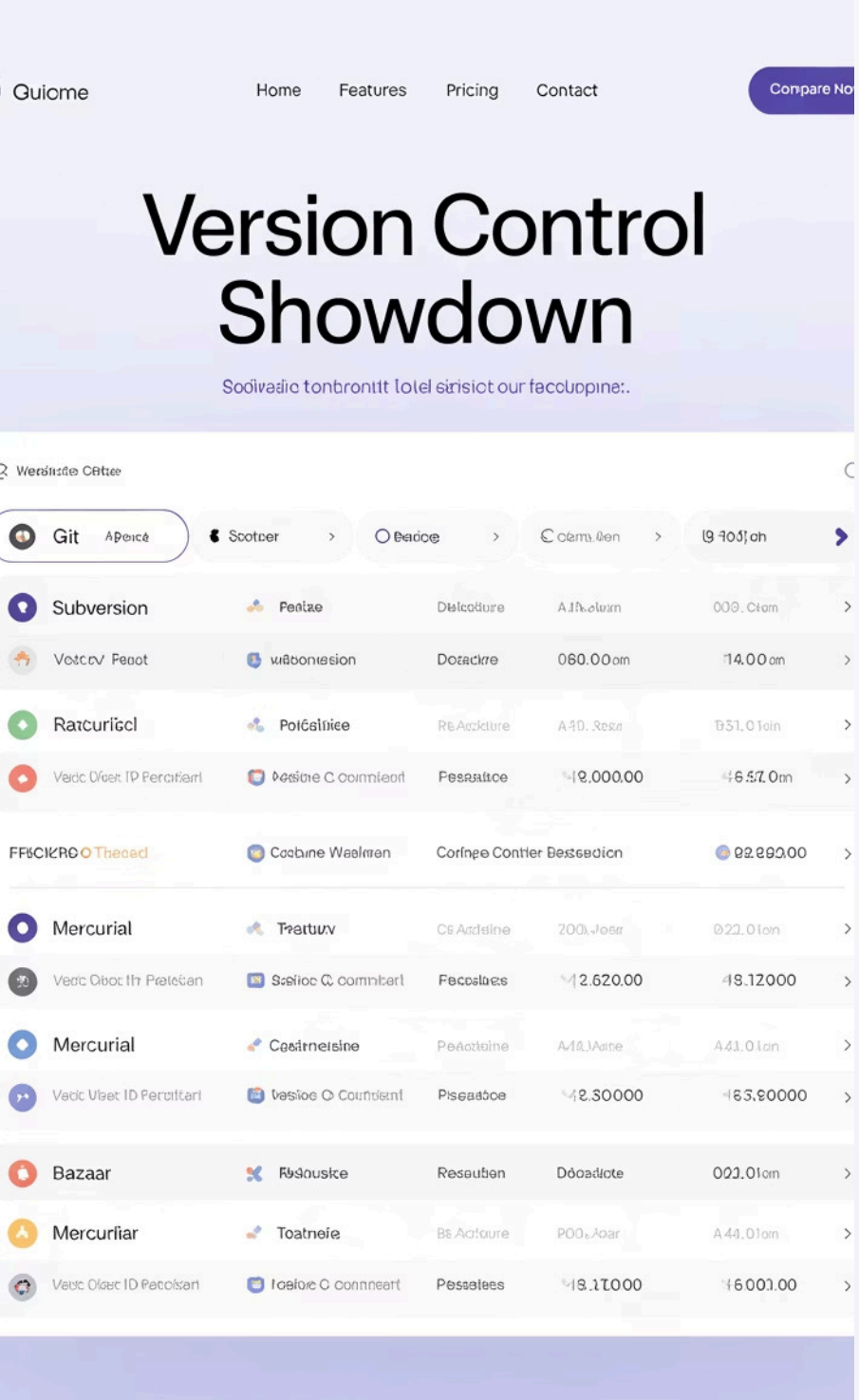
Link remote repositories

3

Version Control

Commit, push, pull operations

Supports R Markdown, EDA scripts, and Shiny application pipeline versioning.



Summary Table

Topic	Why It Matters
Git/GitHub	Reproducibility and collaboration
SourceTree	GUI for intuitive version control
Git Workflow	Structure and consistency
Git Commands	Full versioning control
Branching	Safe experimentation
Push/Pull	Sync and share updates
GitHub Push	Centralized and auditable codebase
Git Software	Industry-wide adoption
Git in R	Track and share analytical work

Key Takeaways

100%

Reproducibility

Every analysis fully traceable

10x

Collaboration

Faster team development cycles

0

Lost Work

Complete change history protection

Adopt Git, GitHub & SourceTree

> From EDA to dashboards, version control matters. Elevate your data-driven workflow today.

Start your journey toward better collaboration, reproducibility, and data governance.

#DataDriven #VersionControl #Collaboration #Excellence