

2018 SGCI Hackathon Training Seminar


If you have not yet joined the SCGI Hackathon Slack Channel, you are missing out!

<https://sgcihackathon.slack.com/>



2018 SGCI Hackathon Training Seminar

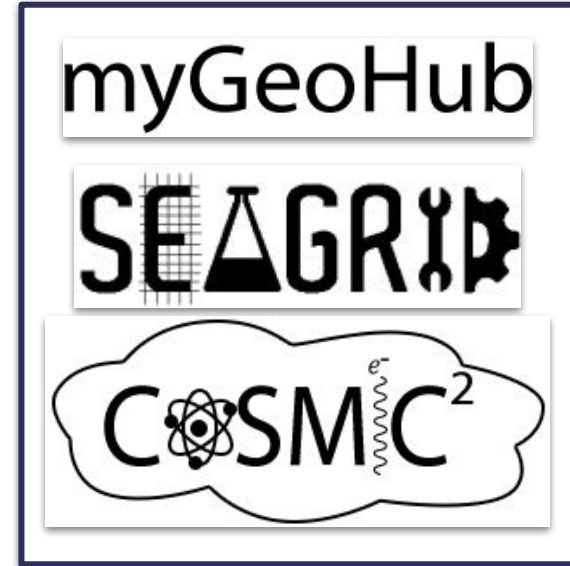
July 10, 2018



Science Gateways Introduction

Presenting: Dr. Linda Hayden - Elizabeth City State University

- **myGeoHub**
 - *Mentor:* Rajesh Kalyanam - Purdue University
- **SeaGRID**
 - *Mentor:* Sudhakar Pamidighantam - Indiana University
- **Cosmic²**
 - *Mentor:* Mona Wong - San Diego Supercomputer Center



Hackathon Judging Criteria

Presenting: Je'aime Powell - Texas Advanced Computing Center

Deliverables:

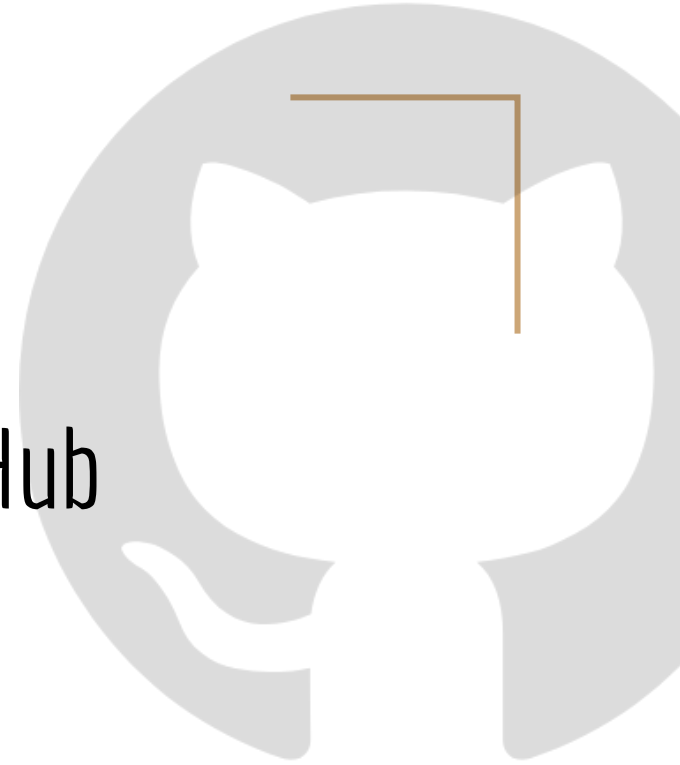
(Posted to Group GitHub Repository)

- Source code
 - Include Comments
- PDF of presentation
- README.md project description

Scored Areas:

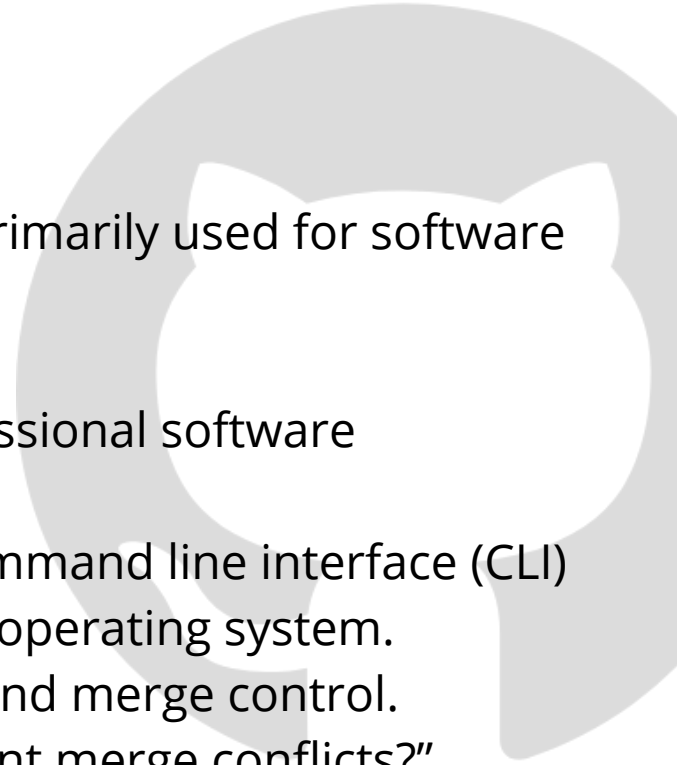
- Visibility / Usefulness (20 pts.)
- Creativity of execution (20 pts.)
- UX / Polish (10 pts.)
- Technical complexity (20 pts.)
- Collaboration (20 pts.)
- Presentation (10 pts.)

Introduction to GitHub



What is GitHub?

- Distributed document version control system primarily used for software development.
- Purchased by Microsoft June 8th, 2018.
- Often viewed by potential employers as a professional software development portfolio for potential employees.
- Most commonly interfaced through the “git” command line interface (CLI) in terminal or command prompt depending on operating system.
- The web-interface does allow basic file editing and merge control.
- Most common issue: “How do I resolve document merge conflicts?”



GitHub Terminology

- **Repository** - A local or remote set of files/code initialized with git
- **Master** - Default/production set of files/code
- **Branch** - *(Also known as a feature)* A snapshot of the master files/code tagged such that development can take place without affecting the master files/code.
- **Commit** - *(Also known as staging)* Alerts git that files/code have been altered, and checks the files against the current master/branch for merge issues.
- **Pull/Push** - Retrieves remote, or sends local files/code changes
- **Merge** - Manual combination of any feature branches and the master branch

GitHub General Flow

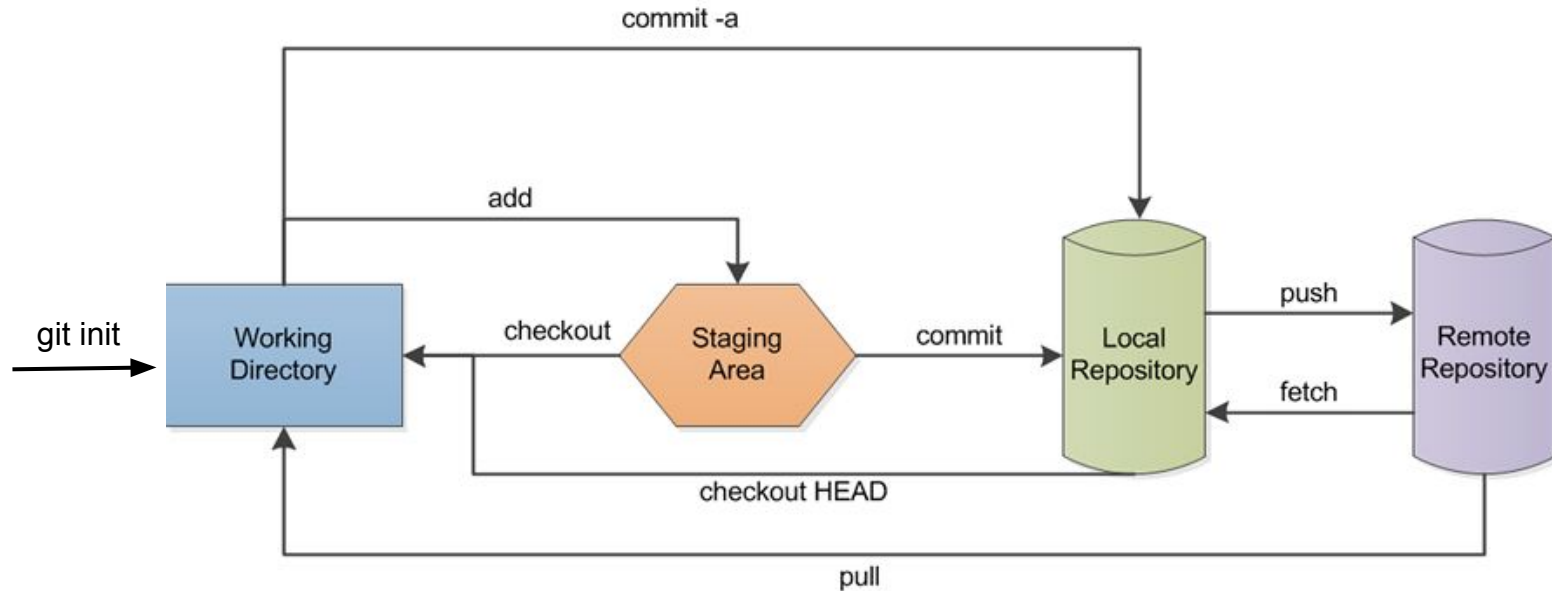


Image Reference: <https://hackaday.com/2017/05/11/history-of-git/>

DEMO TIME

- Create a local Working Directory
- Initialize local git repo
- Check status
- Create a commit (stage a change)
- Push local repo to GitHub
- Branch a feature
- Change / Push / Merge
- Pull new features

Commands Used

- `git clone <repo.url>`
- `git branch <feature>`
- `git checkout <feature>`
- `git add .`
- `git commit -m "reason"`
- `git push origin <feature>`
- `git checkout master`
- `git pull`

Note: There are many ways to do this process. Find one you are comfortable with, and do not be afraid to test (pre-hackathon of course!)

Next Seminar July 17th @ 2:00PM CST

Featured Topic:

Project introductions by the group mentors.

If you have not yet joined the SCGI Hackathon Slack Channel, you are missing out!

<https://sgcihackathon.slack.com/>



Science Gateways
Community Institute

sciencegateways.org/engage/hackathon