GitHub Group Project Overview

Overview: Understanding GitHub and Collaborative Skills

This GitHub Group Project is designed to introduce first-semester computer science students to the fundamentals of version control and collaborative coding using GitHub, an essential tool in the professional programming world. As part of this course, students will not only become familiar with managing repositories but also with working effectively in teams—skills crucial for success in future projects throughout the computer science undergraduate program at the University of Colorado Boulder.
Students will configure their GitHub profiles, set up repositories, commit code, and resolve conflicts in a group setting. By the end of this project, students will have a solid grasp of GitHub's technical aspects, group collaboration dynamics, and conflict resolution, which are critical for both academic and professional success.

Learning Objectives

1. Develop technical proficiency in GitHub: Students will configure GitHub profiles, set up repositories, and perform essential tasks such as commits, pull requests, and conflict resolution.
2. Enhance teamwork and collaboration: Students will collaborate with peers using GitHub in a group setting, learning to communicate effectively and share responsibilities in real-time.
3. Understand version control principles: Students will learn key version control concepts such as cloning repositories, committing changes, resolving conflicts, and managing team workflows.
4. Reflect on group dynamics and individual contributions: Students will engage in peer reviews, reflecting on their team members' and their own performance, offering constructive feedback and identifying areas for improvement.

[All instructions for the project are outlined on this repository.]

Weeks 6-10: Preparatory Mini Activities (this will cover the GitHub section of your narrative for weeks 6-10)

Week 6: Introduction to Version Control and GitHub

Task:

- **How to Install GitHub:**
  - **Windows:** Installing GitHub on Windows
  - **Linux:** Installing GitHub on Linux
  - **Mac:** Installing GitHub on Mac
- **Watch:** The video on version control and GitHub (Version Control Basics) - https://www.youtube.com/watch?v=oZpKSd8wdZk
- **Read:** GitHub Guides - Understanding Version Control - https://github.blog/developer-skills/programming-languages-and-frameworks/what-is-git-our-beginners-guide-to-version-control/

Activity Instructions:

- Create a GitHub account.
- Create a new repository called **IntroToGitHub**.
- Write a brief description of what version control is in the repository's description field.

Submission:

Submit the link to your GitHub repository in the GitHub section of your narrative for this session. Also add a 50 word reflection on the process and key takeaways.

Week 7: Learning Basic Git Commands and Local Repositories

Task:

- **Review:** The guide on basic Git commands: Git Cheat Sheet.

Activity Instructions:

- Install Git on your local machine.
- Initialize a new local repository using `git init`.

- Add a file to your repository and commit it using Git.
- Push the commit to GitHub.

Submission:

Submit a screenshot showing the terminal commands and the successful commit. Include the basic Git commands used (`push`, `add`, `commit`, `pull`, `stash`, `checkout`, `branch`, `merge`) in the GitHub section of your narrative for this session. Also add a 50 word reflection on the process and key takeaways.

Week 8: Configuring GitHub Profiles and SSH Keys

Task:

- **Read:** GitHub guide: Connecting to GitHub with SSH.
- **Reasoning:** Understand why SSH is used and when HTTPS may not be suitable.

Activity Instructions:

- Set up your GitHub profile.
- Configure SSH keys on your local machine.
- Push a change to your repository using SSH.

Submission:

Submit a screenshot of your GitHub settings showing the SSH setup and terminal confirmation of your push in the GitHub section of your narrative for this session. Also add a 50 word reflection on the process and key takeaways.

Week 9: GitHub Issues and Pull Requests

Task:

- **Watch:** The tutorial on GitHub Pull Requests and Issues. Learn about open source development and why pull requests (PRs) are crucial in open-source projects.

Activity Instructions:

- Create an issue in your repository for a small fix or improvement.
- Submit a pull request to address the issue and merge it into the main branch.

Submission:

Submit a screenshot of the issue, pull request, and merge in GitHub in the GitHub section of your narrative for this session. Also add a 50 word reflection on the process and key takeaways.

Week 10: Writing a README File Using Markdown

Task:

- **Review:** The basics of Markdown using the Markdown Guide.

Activity Instructions:

- Create a `README.md` file in your repository.
- Include sections like the project title, description, installation instructions, and usage.
- Add multiple Markdown elements (e.g., headings, lists, images) to your README.

Submission:

Submit the link to your repository with the completed `README.md` file in the GitHub section of your narrative for this session. Also add a 50 word reflection on the process and key takeaways.

Weeks 11-13: GitHub Group Project

Part 1: Week 11 Task (4 points)

- Task:
    - Use Canvas to form groups.
    - Connect with your group using the discussion forum or other means.
    - Configure your GitHub profiles if needed.
    - Set up a remote GitHub repository for your group.
    - Ensure all team members clone the repository and familiarize themselves with steps 1-5 of the GitHub repository.
- Submission:
    - Post a screenshot showing your group's repository and initial setup in the assignment area.
    - Add a 50 word reflection on the process, group dynamics and collaboration, and key takeaways.

Part 2: Week 12 Task (4 Points)

- Task:
    - Add personal pages to the group project.
    - Commit your changes to the shared GitHub repository.
    - Create changes locally, take a pull after the changes, then resolve the conflicts.
- Submission:
    - Post a screenshot of your progress (personal pages added) in the assignment area.
    - Add a 50 word reflection on the process, group dynamics and collaboration, and key takeaways.

Part 3: Week 13 Task (4 Points)

- Task:
    - Reflect on your group's overall experience using GitHub.
    - Write a 50-word reflection on your group's progress and any challenges encountered.
- Peer Review Component:
    - Evaluate each team member's contribution and performance.
    - Criteria: participation, contribution, communication, deadlines.
    - Provide constructive feedback on each team member's performance in a private peer review form.
- Submission:
    - Post a screenshot showing your group's final repository, including your commits.
    - Submit a link to your group's repository.
    - Submit the peer review form and a 50 word reflection of key takeaways form the project overall and a warm critique of your own performance.

GitHub Project Rubric

Part 1: Week 11 Task (4 Points Total)

Criteria	Exemplary (1 point)	Satisfactory (0.5 points)	Needs Improvement (0 points)
Group Formation and Communication	All group members are formed, connected, and communicated efficiently via the discussion forum or other means.	Group members are mostly formed and communicated, but with minor issues or delays.	Group formation or communication was incomplete or ineffective.
GitHub Profile Configuration	All group members' GitHub profiles are configured correctly, with no issues.	Most group members have configured their GitHub profiles, but some minor issues remain.	GitHub profiles were not configured appropriately or not at all.
Repository Setup and Cloning	Group successfully sets up a remote repository and all members clone the repository.	Repository setup is mostly successful, but some minor issues exist, such as a missing step or a delayed cloning process.	Repository setup or cloning was incomplete or not done.
Screenshot Submission	The screenshot is clear and shows the entire group's repository setup as required.	The screenshot is submitted but lacks clarity or does not show the full repository setup.	No screenshot submitted, or the screenshot is incorrect.

Part 2: Week 12 Task (4 Points Total)

Criteria	Exemplary (1 point)	Satisfactory (0.5 points)	Needs Improvement (0 points)
Personal Pages Added	Personal pages added with clean, clear formatting and no errors in the content.	Personal pages added, but with some minor formatting or content issues.	Personal pages were not added, or the formatting and content were significantly flawed.

Commit Changes to Repository	Commits are made effectively, and all changes are reflected in the shared GitHub repository.	Commits are made, but minor issues exist, such as missed commits or lack of synchronization with the repository.	No commits made, or commits were incomplete and not reflected in the shared repository.
Conflict Resolution	All conflicts are resolved professionally, and screenshots of each step (local changes, pull, conflict resolution) are submitted.	Conflicts are resolved, but steps are incomplete or missing, such as not providing all necessary screenshots.	Conflicts were not resolved, or screenshots of conflict resolution are not provided.
Screenshot Submission	A clear screenshot is provided, showing personal pages added and changes committed successfully.	The screenshot is provided but lacks clarity or does not fully demonstrate the personal pages or commits.	No screenshot provided, or the screenshot does not accurately show the required progress.

Part 3: Week 13 Task (4 Points Total)

Criteria	Exemplary (1 point)	Satisfactory (0.5 points)	Needs Improvement (0 points)
Group Reflection (50 words)	Reflection includes detailed insights into the group's overall progress and challenges.	Reflection covers basic group progress but lacks depth or critical insight into challenges.	Reflection lacks clarity or is incomplete, with minimal discussion of group progress or challenges.
Peer Review Contribution and Feedback	Thoughtful and constructive feedback is provided to all team members, highlighting contributions and areas for improvement.	Feedback is provided but lacks depth or does not fully address contributions and areas for improvement.	Feedback is minimal or absent, with no constructive or meaningful contributions.
Final Repository Submission	The final repository is complete, functional, and all required commits and contributions are present.	The final repository is mostly complete, but some commits or contributions are incomplete.	The final repository is incomplete, with missing commits or contributions from group members.

Screenshot Submission	A clear screenshot is provided, showing the final repository and individual contributions.	The screenshot is provided but lacks clarity or does not fully demonstrate the final repository.	No screenshot provided, or the screenshot is incorrect or missing required information.