Group Coursework – Portfolio Website Project

Course: Version Control System (VCS)  
Submission Type: Group Project  
Group Size: 3–5 Students  
Project Duration: 3 Weeks

# Project Overview

This coursework is designed to evaluate your collaborative development skills, Git/GitHub proficiency, and ability to reflect your learning through a live, visually engaging portfolio website.  
  
You will work in a group to design and build a minimum 6-page website with an integrated Git workflow. The website will also reflect content and design principles from your UI/UX course. Each member’s portfolio must be integrated within an 'About Us' page.

# Tutorial Session: Project Proposal

During your tutorial session, your group must create and submit a project proposal for approval.

## Proposal Template

Each group must submit a short proposal (1–2 pages) including the following:  
1. Introduction – Brief description of the project idea  
2. Unique Project Name – A creative and relevant name  
3. Reason – Why your group chose this idea  
4. Learning Reflection – What learning aspects you will showcase  
5. Visual Plan – How you will represent learning through the website

# Project Requirements

## Website Requirements

A minimum of 6 functional pages:  
 • Home (mandatory)  
 • About Us (integrated portfolios)  
 • A form-based page (e.g., contact or feedback)  
 • 3 or more pages based on your topic (e.g., Gallery, Projects, Blog, Services)  
- UI/UX must align with principles taught in your design course  
- Real or placeholder data and images  
- Responsive and aesthetically consistent design  
- You may use HTML5 UP or similar templates as a base

## Git & Version Control Requirements

- Use Git and GitHub for the full development lifecycle  
- Each member must use a feature branch for their tasks  
- Perform regular commits with clear messages  
- Use pull requests for merges  
- Resolve conflicts manually at least once  
- Maintain a clean and organized commit history  
- Add all collaborators with correct GitHub permissions  
- One designated admin should control the main branch  
- Host the site on GitHub Pages or a similar platform

Submit the following Git evidence:  
- Screenshots of: push, pull, clone, merge  
- `git log`, `git branch`, `git log --graph` output  
- Branch graph visual  
- Merge conflict screenshot  
- Contributions visible in GitHub insights

# Final Deliverables

1. GitHub Repository Link (with all collaborators added)  
2. Live Hosted Website Link (GitHub Pages/Netlify/Vercel)  
3. Final Report (max 10 pages):  
 • Overview of the project & team roles  
 • Git activities with screenshots  
 • Summary of Git commands used  
 • Merge conflict documentation  
 • Team contribution breakdown  
 • Reflection on Git usage  
4. README.md (on GitHub):  
 • Project description  
 • Setup instructions  
 • Contributors with GitHub profiles  
 • Screenshots and/or demo video link

# Assessment Criteria

|  |  |
| --- | --- |
| Criteria | Marks |
| Website functionality and UI/UX alignment | 20% |
| Proper Git workflow (branches, commits) | 25% |
| GitHub repository quality (README, structure) | 15% |
| Group collaboration and Git log tracking | 15% |
| Website hosting and access | 10% |
| Final Report and reflections | 15% |

# Tips & Guidelines

- Start early and plan your Git strategy.  
- Assign pages and responsibilities clearly.  
- Use GitHub issues or a project board for task tracking.  
- Make your commits meaningful and focused.  
- Don’t push to `main` directly; use pull requests.  
- Align your design with your UI/UX coursework.  
- Communicate regularly within your group.  
- Submit on time with complete documentation.