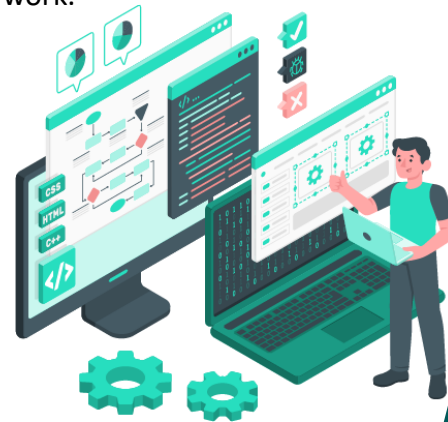


PRODUCT DEVELOPMENT FOR SCIENTIFIC GUI APPLICATIONS

Exploring user-centric design and data analysis for geophysical work.



Presented by

Harsh Prajapati
M.Sc.Tech
Applied Geophysics

Ritesh Gond
M.Sc.Tech
Applied Geophysics

Guided by:
Prof. Partha Pratim Mandal

AGENDA

Exploring the GUI Development in Scientific Applications.

1 Introduction

2 Development of Workflow

3 Innovative Ideas & Problem Identification

4 Programming Essentials

5 Git and GitHub

6 PyQt GUI Development

7 Conclusion

INTRODUCTION

Importance of Product Development in Scientific Contexts.

- **Enhancing data analysis through interactive applications.**

Interactive applications offer dynamic data visualization and manipulation features, allowing researchers to explore and interpret data more effectively.

- **The need for intuitive tools in scientific fields.**

Scientific applications require user-friendly tools for efficient data processing and analysis, improving overall productivity and user experience.

- **Example: Petrophysical package project.**

Explore PETROHARRIT and LASMasterEditor projects as showcases of successful product development efforts in scientific applications, demonstrating the impact of intuitive tools.



DEVELOPMENT OF WORKFLOW

Understanding the process from coding to testing is crucial for efficient development.

➤ Design the Workflow

Develop a detailed workflow from data loading to parameter calculation, outlining each step and its sequence.

➤ Create a User-Friendly Interface

Design an intuitive interface, incorporating innovative ideas and iterating based on user feedback to address usability issues effectively.

➤ Implement and Maintain

Develop modular code for each workflow step and UI component, following best practices to ensure maintainability, reusability, and generalizability.



INNOVATIVE IDEAS & PROBLEM IDENTIFICATION

➤ Understanding the Problem Space

In petrophysical analysis, issues like data inconsistency, complex calculations, and visualization needs are common pain points.

➤ Prioritizing Problems

Not all problems are equal—focus on those with the highest impact on the user's workflow.

➤ Creating a Problem Statement

A well-defined problem statement should address the who, what, and why.



Programming Essentials

➤ Learn Python Basics and Tools

- Understand control structures, data types, and OOPs.
- Use IDEs like VS Code or PyCharm.
- Use .py files and basic interpreters.

➤ Understand Packages and Event-Driven Programming

- Learn Python package structure and event handling for GUIs.

➤ Develop with PyQt and Qt Designer

- Use PyQt for desktop apps and Qt Designer for easy GUI design.



GIT and GitHub

1. Version Control and Collaboration

- Track changes in your code with a detailed history of edits.
- Revert to previous versions if needed, reducing risks of errors.
- Work simultaneously with others without overwriting each other's changes.

2. Branching and Merging

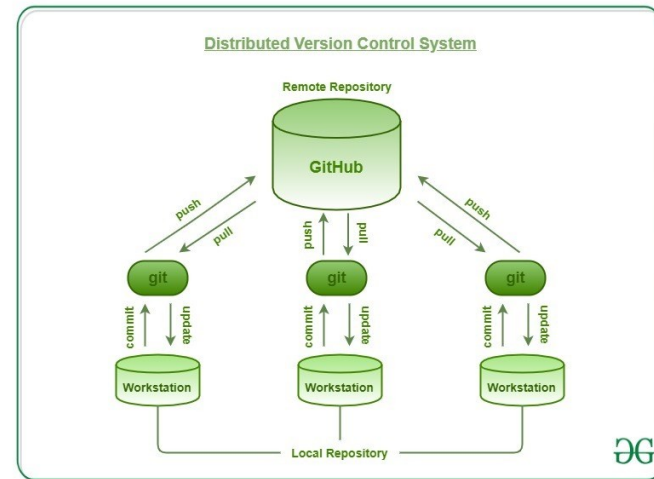
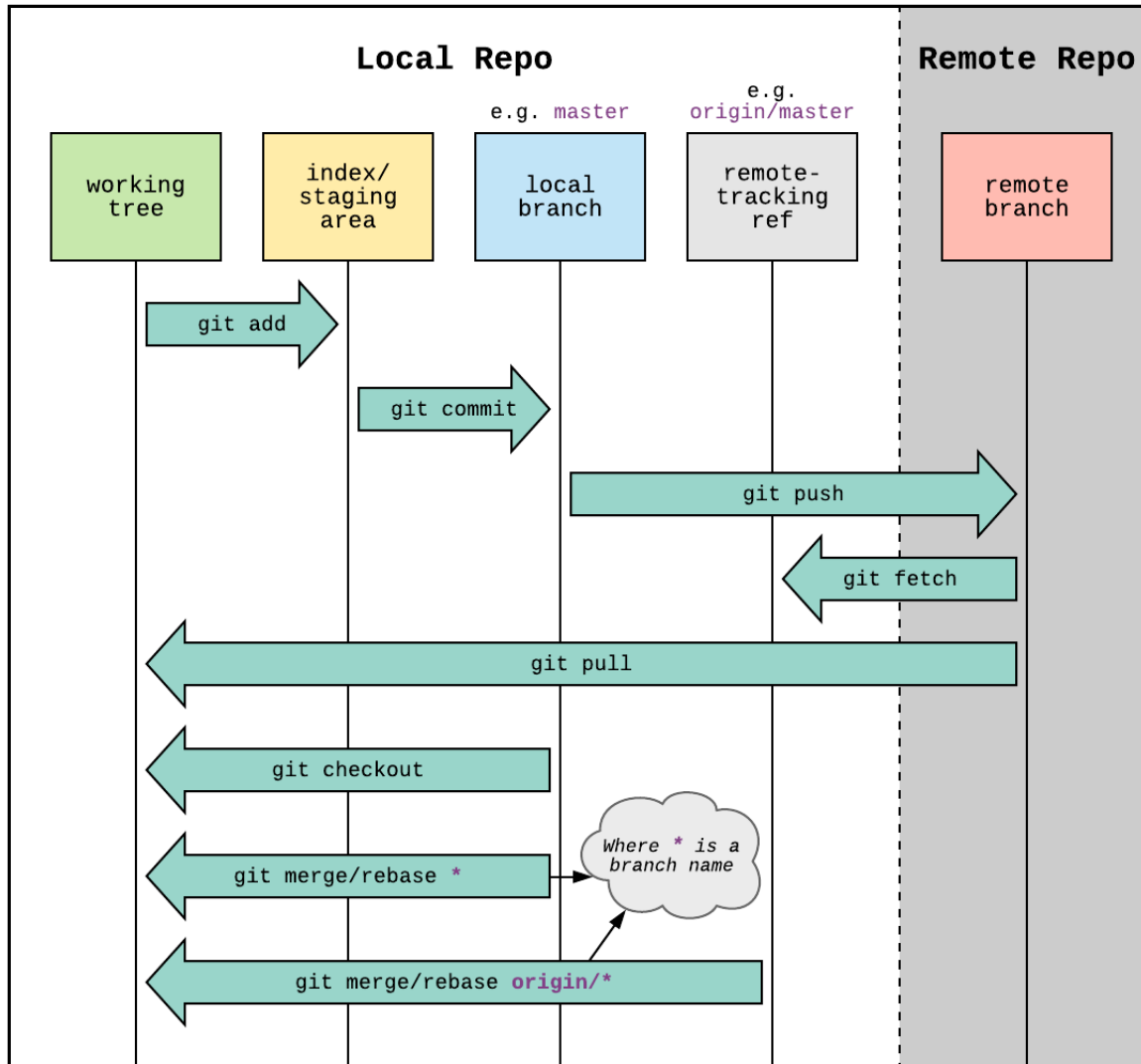
- Create branches for new features or bug fixes, isolating your work from the main codebase.
- Merge branches when ready, integrating new changes smoothly.
- Experiment with different ideas without affecting the main project.

3. GitHub Integration




- Host your repositories online with access control for team collaboration.
- Use pull requests to propose, review, and discuss changes before merging.
- Manage tasks and issues directly on GitHub, keeping project organization in one place.



GIT and GitHub



GIT and GitHub

- 1) Create a  GitHub account.
- 2) Download  git on your device and log in.
- 3) Follow codes in  command prompt:
 - i. `git init`
 - ii. `git remote add origin <repo link>`
 - iii. Write your codes
 - iv. `git add <file name>`
 - v. `git commit -m "Your update message"`
 - vi. `git push -u origin <branch>`



PyQt GUI Development

1. **Creating a Virtual Environment**
2. **Installation of PyQt5**
3. **Introduction to PyQt5's Modules**
4. **Window (Container) Widgets**
5. **Layouts in PyQt**
6. **Basic Widgets**
7. **PyQt with Matplotlib**
8. **Qt Designer**



CONCLUSION

➤ Start Your Journey

Apply the concepts learned today—focus on creating intuitive GUIs and innovative solutions in Python for geophysical applications.

➤ Embrace Challenges

Solve real problems, improve workflows, and keep iterating on your designs. Innovation and persistence will set your product apart.

➤ Solving Problem Statement

Continuously explore, learn, and code. Your unique ideas can drive advancements in scientific applications.



**THANK
YOU**