



Search OOP

VERSION CONTROL SYSTEM

Presented by Group IV



OBJECTIVES

- 1 What Is VCS?
- 2 Version Tracking
- 3 Collaborating
- 4 Branching & Merging
- 5 Backup
- 6 Audit Trail
- 7 Types of Version Control Systems
- 8 Why Use a Version Control System?
- 9 Our Simple DVCS
- 10 Conclusion





WHAT IS VCS?

A Version Control System (VCS), also known as a Code Control System, is a tool or system that helps developers manage changes to source code over time. It keeps track of every modification made to the code, enabling teams to collaborate efficiently, revert to earlier versions, and resolve conflicts.

Key Features of Version Control Systems:

- Version Tracking
- Collaboration
- Branching and Merging
- Backup
- Audit Trail



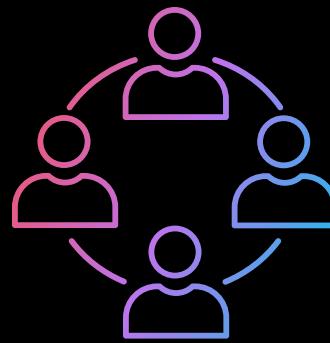


KEY FEATURES OF VERSION CONTROL SYSTEM



Version Tracking:

- Records every change made to the codebase, including who made the change and when
- Allows developers to roll back to previous versions if needed



Collaboration:

- Enables multiple developers to work on the same project simultaneously without overwriting each other's changes
- Helps resolve conflicts when different changes are made to the same code



Backup:

- Acts as a backup for the project since all changes are stored in the system



Audit Trail:

- Maintains a history of all changes, making it easier to track the evolution of the code and debug issues

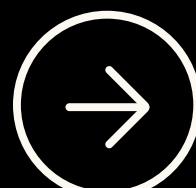


Branching and Merging:

- Allows developers to create branches for new features, bug fixes, or experiments without affecting the main codebase
- Merges branches back into the main codebase once the changes are ready



TYPES OF VERSION CONTROL SYSTEMS



Local Version Control Systems

- All version tracking is done on a single computer

Centralized Version Control Systems (CVCS)

- A single server stores the codebase, and developers pull the code from the server (e.g., SVN, CVS).
- Pros: Simple to manage
- Cons: A single point of failure (if the server is down, no one can collaborate)

Distributed Version Control Systems (DVCS)

- Each developer has a complete copy of the code repository on their local machine (e.g., Git, Mercurial)
- Pros: Better performance, collaboration offline, and no single point of failure

POPULAR VERSION CONTROL TOOLS

GIT

- The most widely used DVCS
- Used with platforms like GitHub, GitLab, and Bitbucket

SUBVERSION (SVN)

- A popular CVCS

MERCURIAL

- Another distributed DVCS

PERFORCE

- Often used in large enterprises

WHY USE A VERSION CONTROL SYSTEM?

- Improves productivity by enabling collaborative coding
- Helps maintain code integrity and reduces the risk of losing work
- Facilitates continuous integration and deployment (CI/CD) processes
- Tracks accountability by showing who made specific changes



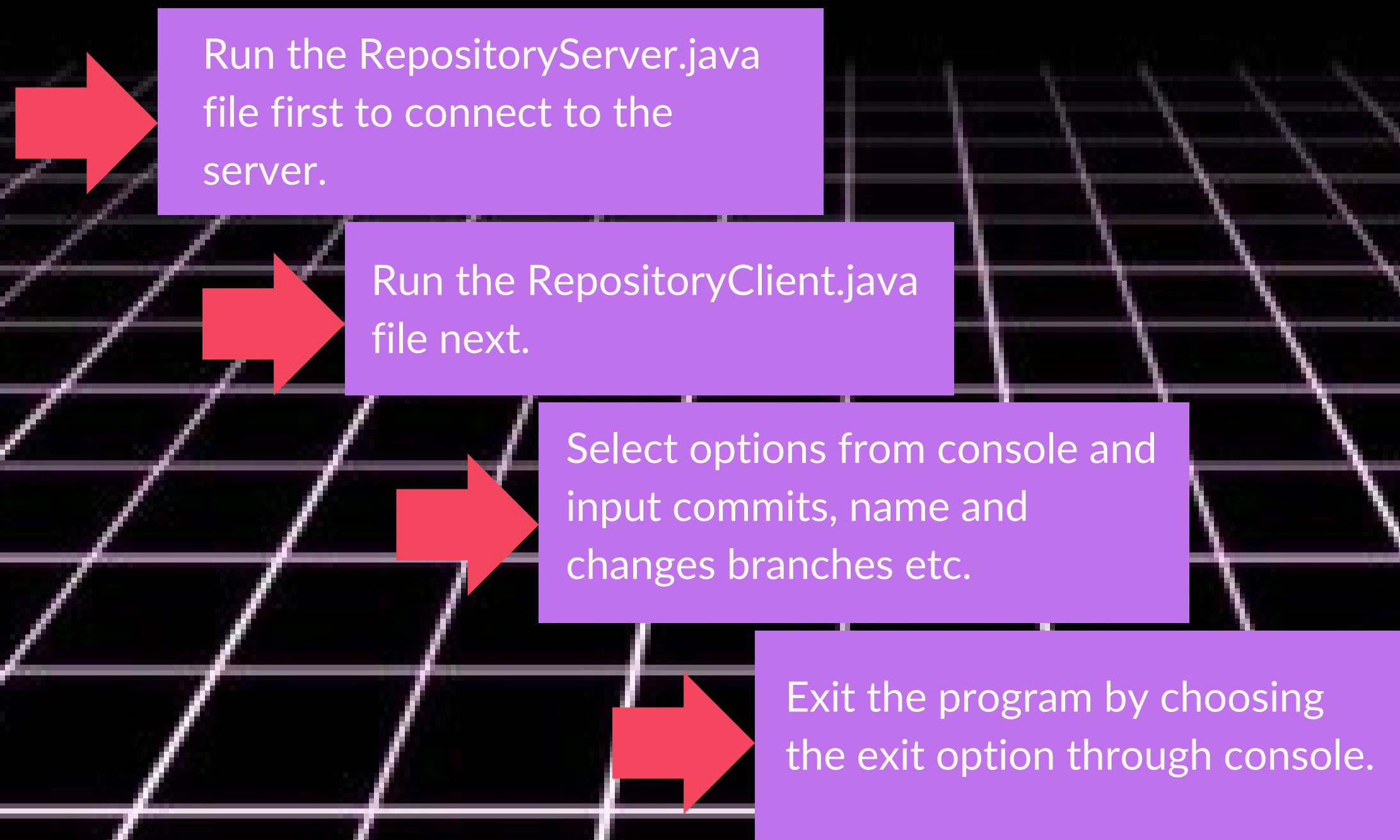
OUR SIMPLE DISTRIBUTED VERSION CONTROL SYSTEM

To represent how a version control system works, we built a simple DISTRIBUTED VERSION CONTROL SYSTEM using the object oriented programming language JAVA. This version control system can operate locally through a local computer while also being connected to the server network. First we need to confirm our server connection system is running in the background while we operate through user/client commits and commands adding the feature of FILE I/O to have the backup of our operating work history.

The whole system is majorly divided into 5 task files that will jot down the whole program:

- **COMMIT** - Mainly built to receive the commit from the user/client.
- **BRANCH** - Built to create the branch name and store the commit inside the branches.
- **REPOSITORY** - Using FILE I/O features to input and output data from user/client from their individual destination/assigned data storage.
- **REPOSITORY SERVER** - Built to connect the user/client to the server of DVCS. This is needed to be run locally at least once to connect the local client's ID address to the server.
- **REPOSITORY CLIENT** - Built to take commands through console from client to operate the system.

KEY EXECUTION STEPS





Version Control System

THANK YOU

for your time and attention

Presented by

1. Nibedita Das - 2411860642
2. Mahabuba Rahman - 2231128042
3. Fairuz Dilshad - 2221740042

