

# Git and GitHub

## Version Control Systems and Teamwork



**SoftUni Team**  
Technical Trainers



**SoftUni**



**Software University**

<https://about.softuni.bg>

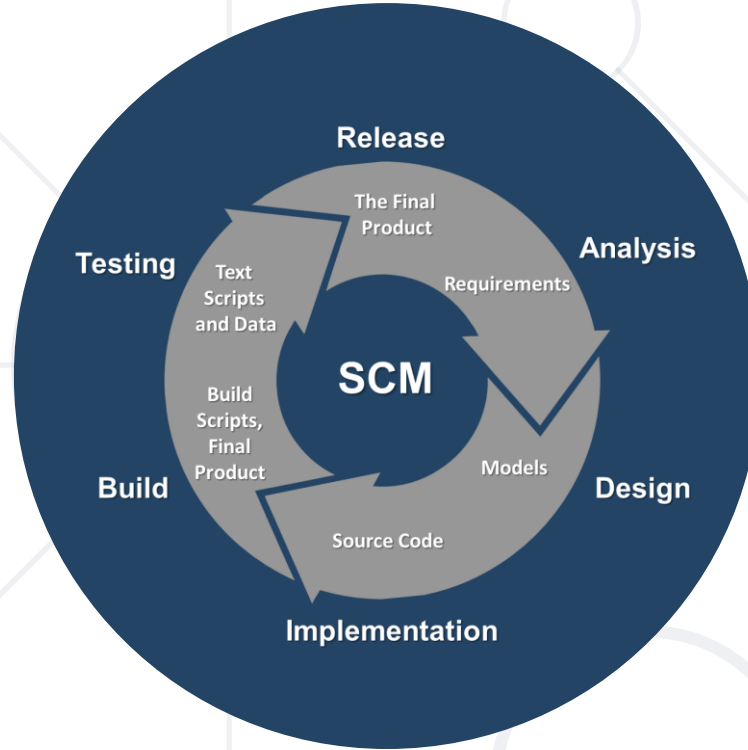
# Have a Question?

**sli.do**

**#qa-fund**

1. Software Configuration Management and **Source Control Systems**
  - Vocabulary: **Clone** a Repo, **Commit** a Changeset, **Push** the Changes, **Pull** Changes, **Merge** Changes
2. Introduction to **Git**
  - Working with Git, Git Bash, and GitHub Desktop
3. Introduction to **GitHub**
  - Create a Repo, Clone, Commit, Push, Conflicts

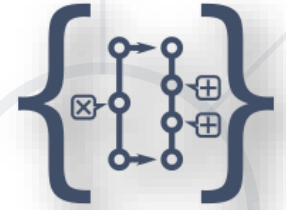




# Software Configuration Management

Working on Shared Code: Source Control Systems

- **Source control systems** keep the source code (+ other project assets) in a shared **repository**
  - Developers can **clone** a repository, **pull** the latest version, **commit** & **push** local changes, view the change logs, etc.
- **Git** is the most popular source control system
  - Other version control systems: SVN, TFS, Perforce
- **GitHub** is the #1 site for Git project hosting
  - Git hosting + issue tracker + review system + project tracker + build system



# Software Configuration Management

- **Version control**  $\approx$  Software Configuration Management (SCM)  $\approx$  **source control system**
  - A software engineering discipline
  - Consists of techniques, practices and tools for working on **shared source code** and files
  - Mechanisms for **changes tracking** and **conflict resolution**
  - Defines the process of **change management**
  - Keeps track of **what is changing** in the project over time
  - Solves **conflicts** in the changes

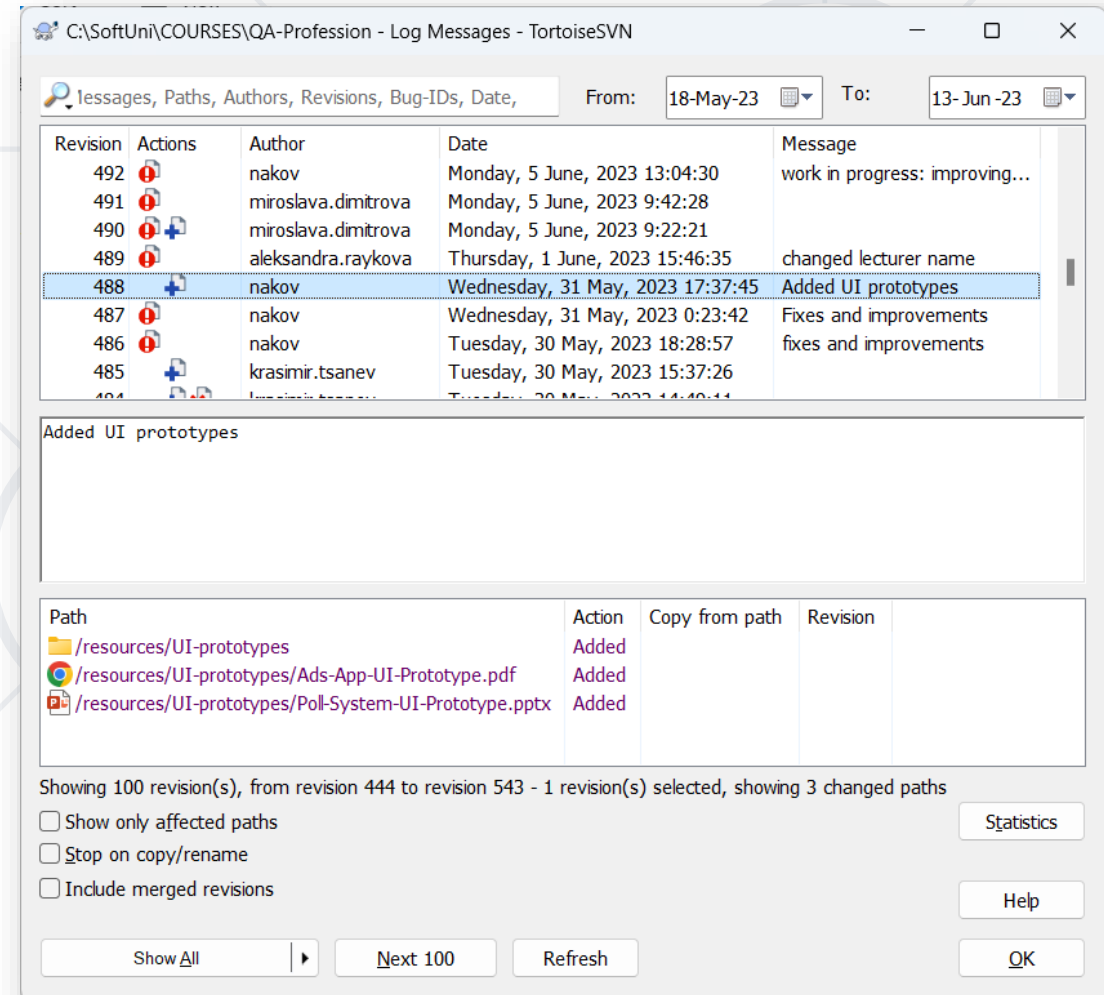


# Change Log

- Version control systems keep their own **change log** (version history):

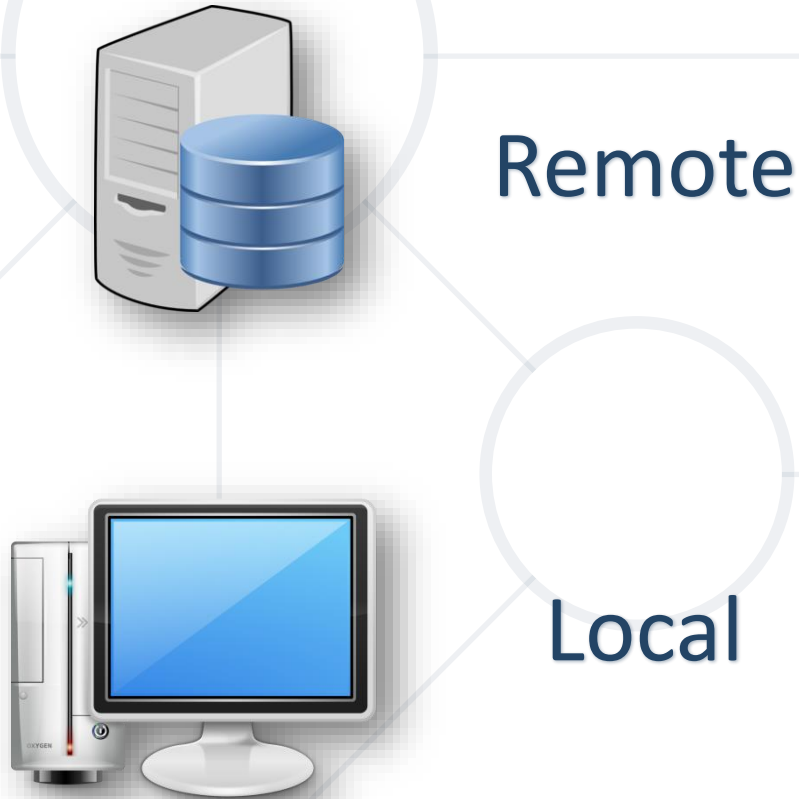
- Who?
- When?
- Why?
- What was changed?

- Old versions can be **reverted back**



# Vocabulary: Repository (Repo)

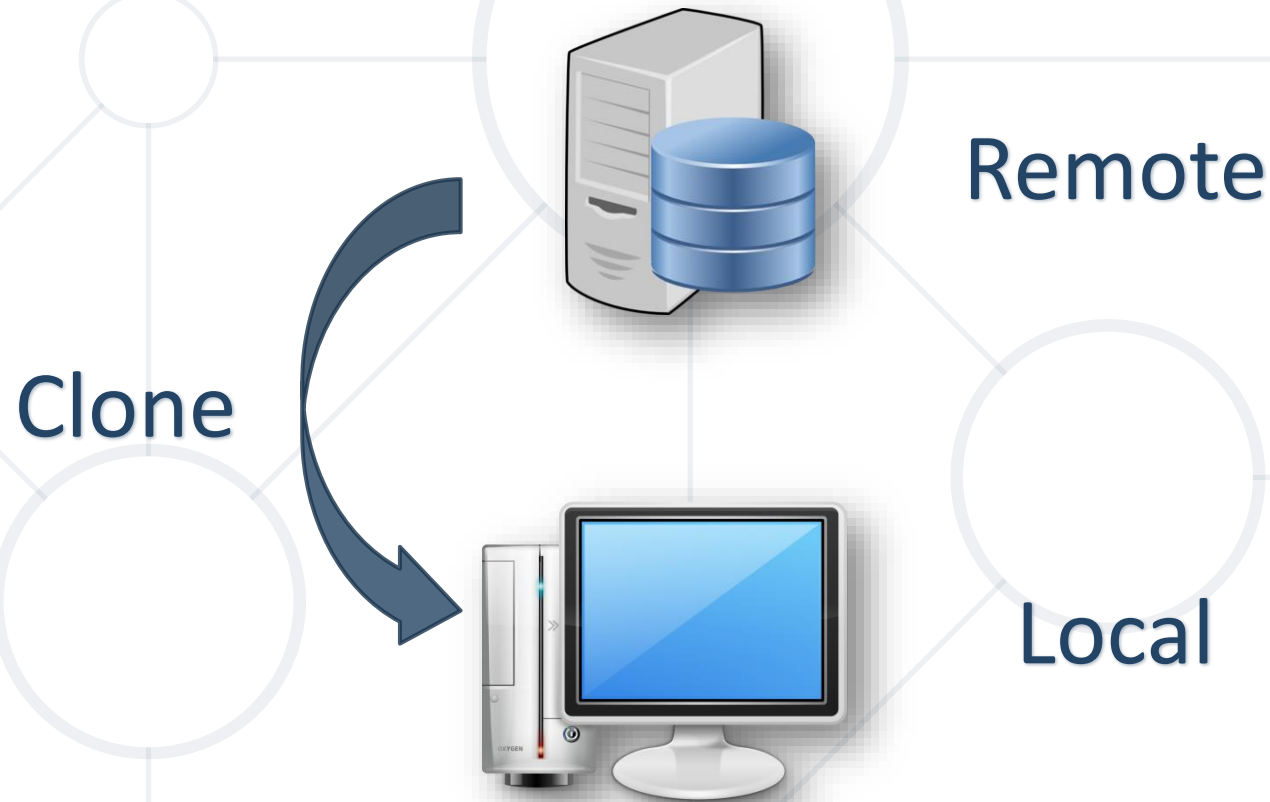
- **Repo** holds the project in a remote server



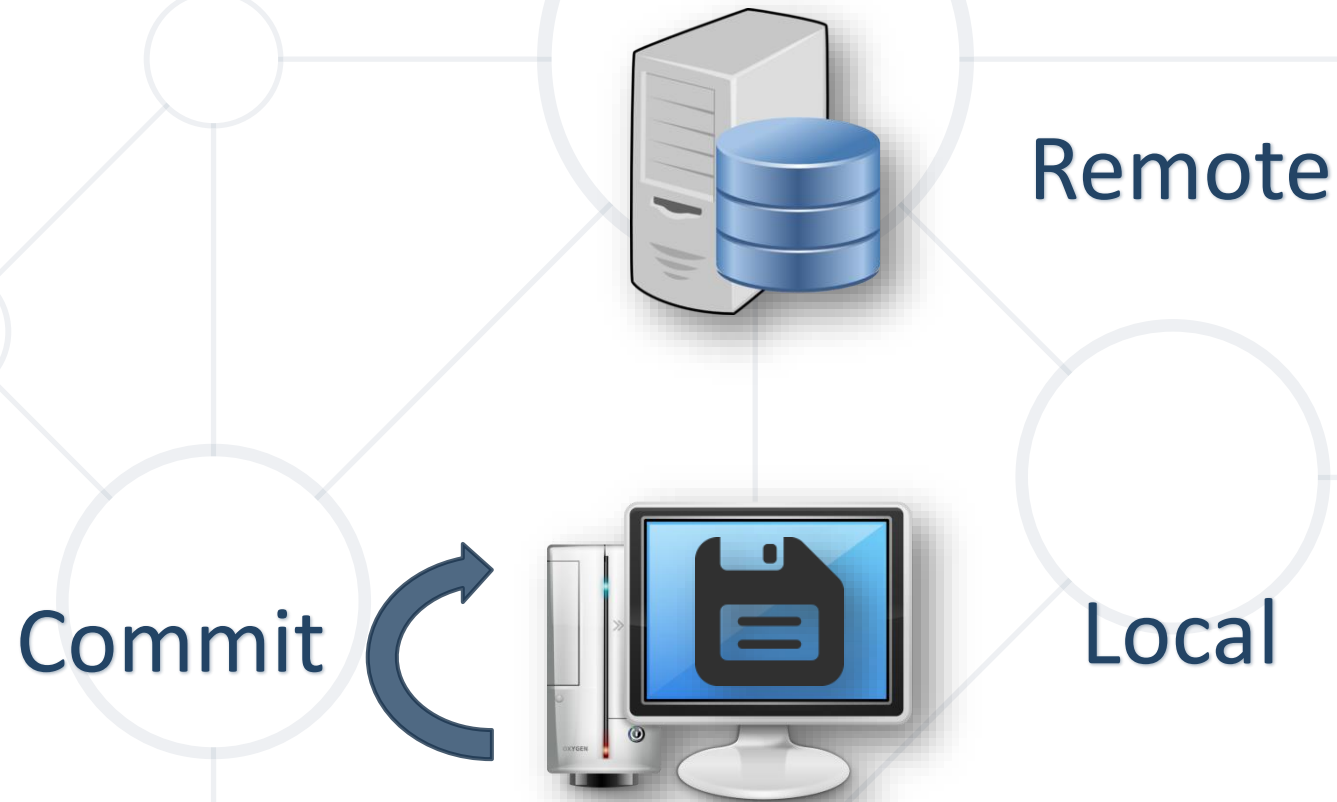


# Vocabulary: Clone

- **Clone** == download a **local copy** of the remote project

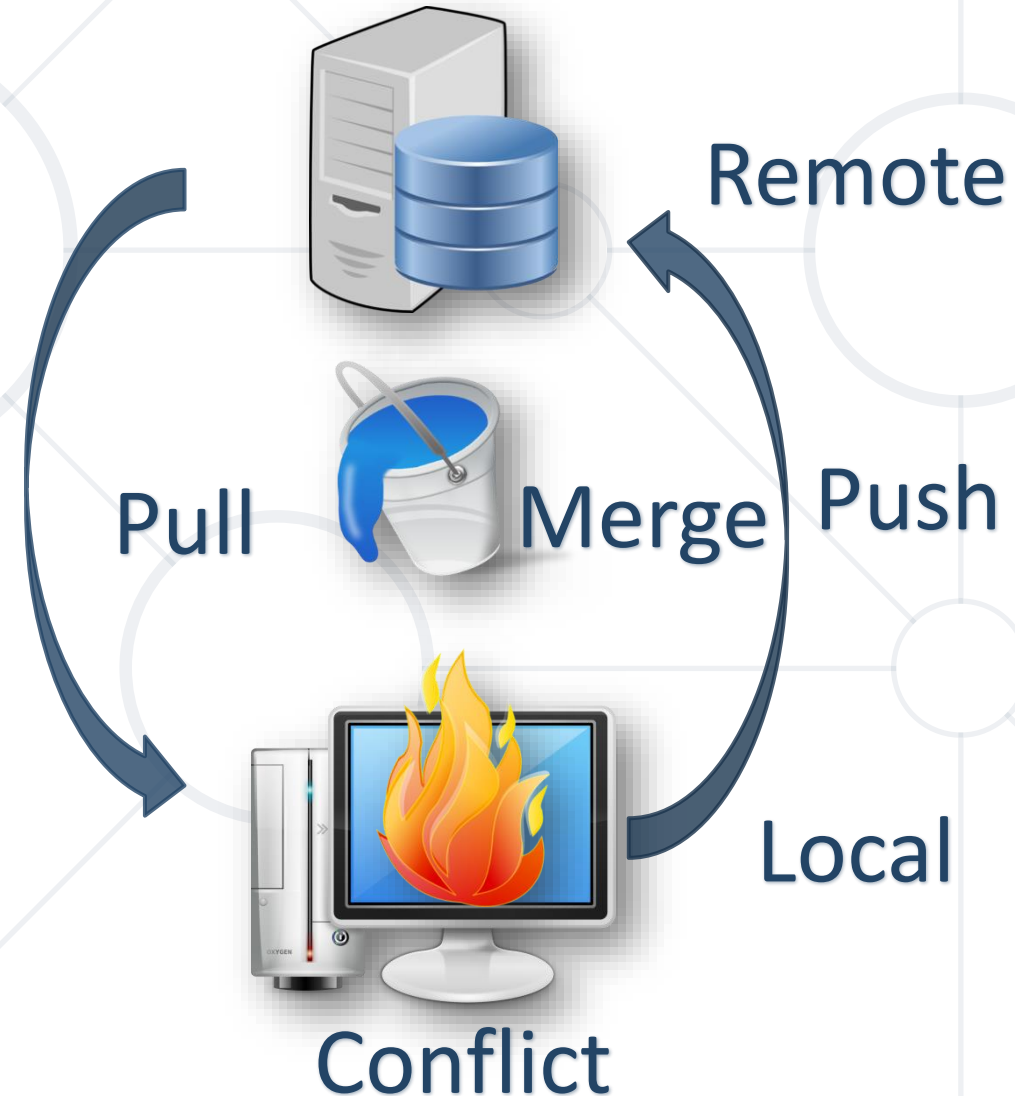


- **Commit** == saves a set of changes locally



# Vocabulary: Sync (Pull / Push)

- **Pull** – take and merge the changes from the Remote
- **Push** – send local changes to the Remote



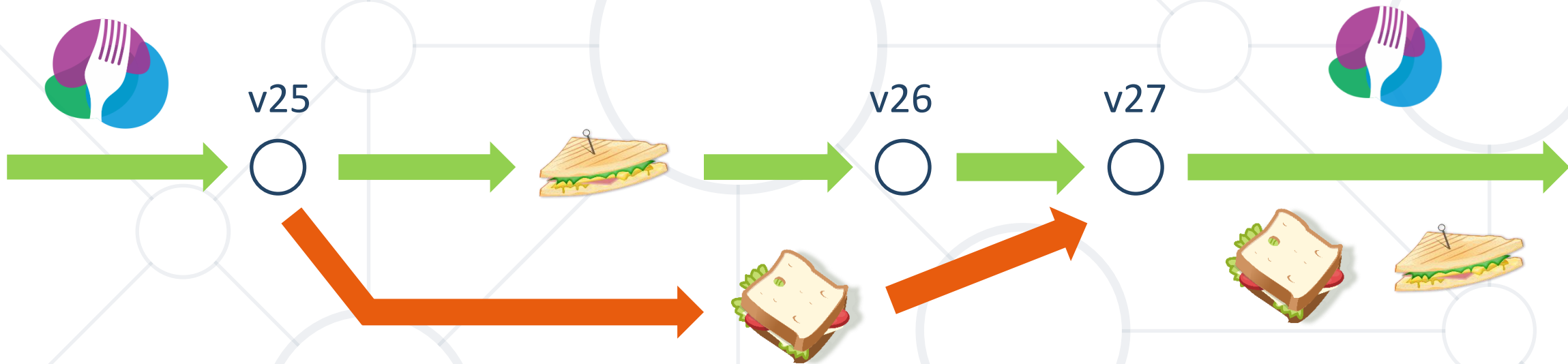
# Vocabulary: Branch



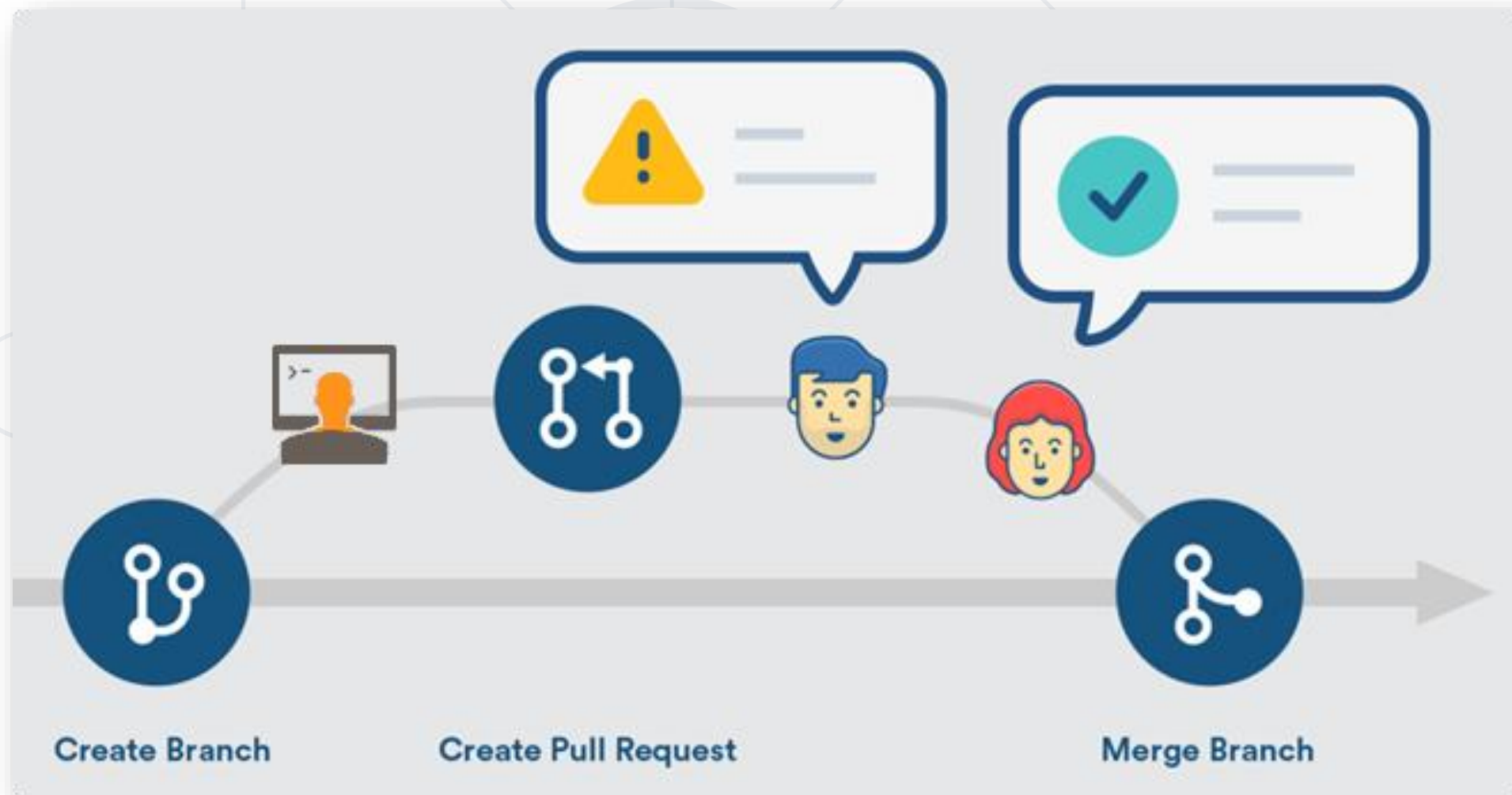
# Vocabulary: Merge Branches



# Example: Branches



# Pull Requests: The Code Review Process





**Git**

World's #1 Source Control System



# What is Git?

- **Git** == distributed **source-control system**
  - The most popular in the world
  - Free, open-source software
- Works with **local** and **remote** repositories
- **Git bash** – command line interface for Git
- Runs on Linux, macOS and Windows (**msysGit**)
- <https://git-scm.com>



# Using Git

- Console-based Git client

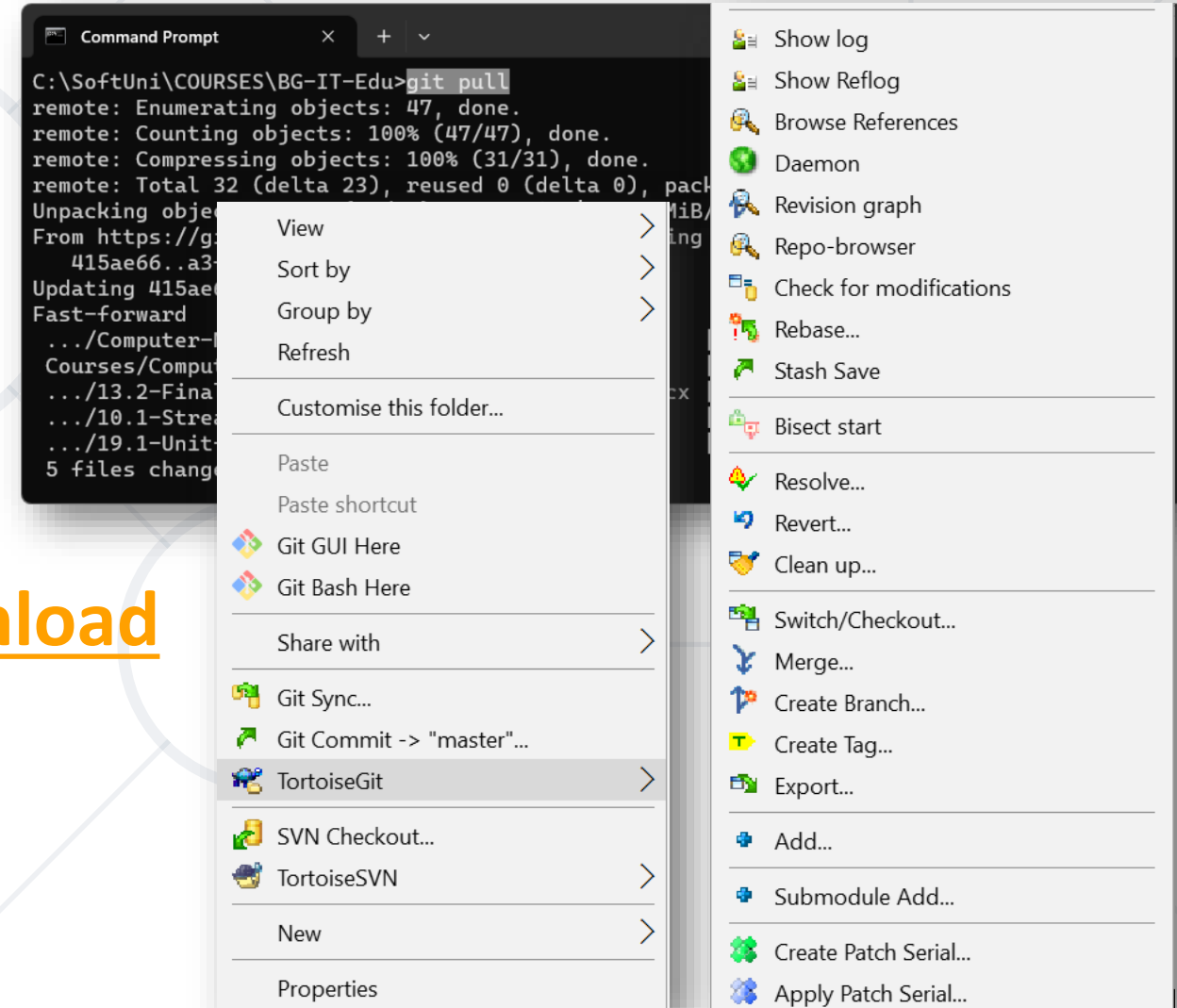
- **git, Git Bash**

- **TortoiseGit** – Windows GUI client

- <https://tortoisegit.org/download>

- **GitHub Desktop** client

- <https://desktop.github.com>



# Installing Git

- Git installation on Windows: **Git for Windows** (msysGit)
  - <https://git-scm.com/downloads>
  - Options to select (selected by default)
    - "Use Git Bash Only"
    - "Checkout Windows-style, Commit Unix-style Endings"
- Git installation on **Linux**:

```
sudo apt-get install git
```



- **Cloning** an existing Git repository

```
git clone [remote url]
```

- **Fetch** and **merge** the latest changes from the remote repository

```
git pull
```

- Preparing (**adding** / selecting) files for a commit

```
git add [filename] ("git add ." adds everything)
```

- **Committing** to the local repository

```
git commit -m "[your message here]"
```

- Check the **status** of your local repository (see the local changes)

```
git status
```

- Creating a **new** local repository (in the current directory)

```
git init
```

- Creating a **remote** (assign a short name for remote Git URL)

```
git remote add [remote name] [remote url]
```

- **Pushing** to a remote (send changes to the remote repository)

```
git push [remote name] [local name]
```

- **Clone** a repository from GitHub

```
git clone https://github.com/SoftUni/playground
```

- **Modify** local files

```
notepad README.md
```

- **Commit** changes (local)

```
git add . & git commit -m "Added something"
```

- **Push** the changes to GitHub

```
git push
```



## **Git**

Checkout → Modify → Commit →  
Push → Resolve a Conflict

# GitHub – Teamwork Example

## ■ Alice (terminal)

```
git clone https://github.com/SoftUni/playground
```

```
notepad README.md
```

```
git add . & git commit -m  
"Added something"
```

```
git push
```

## ■ Bob (TortoiseSVN)

```
git clone https://...
```

```
notepad README.md
```

```
git add & git commit
```

```
git push → conflict
```

```
git pull → merge
```

```
git push → success
```





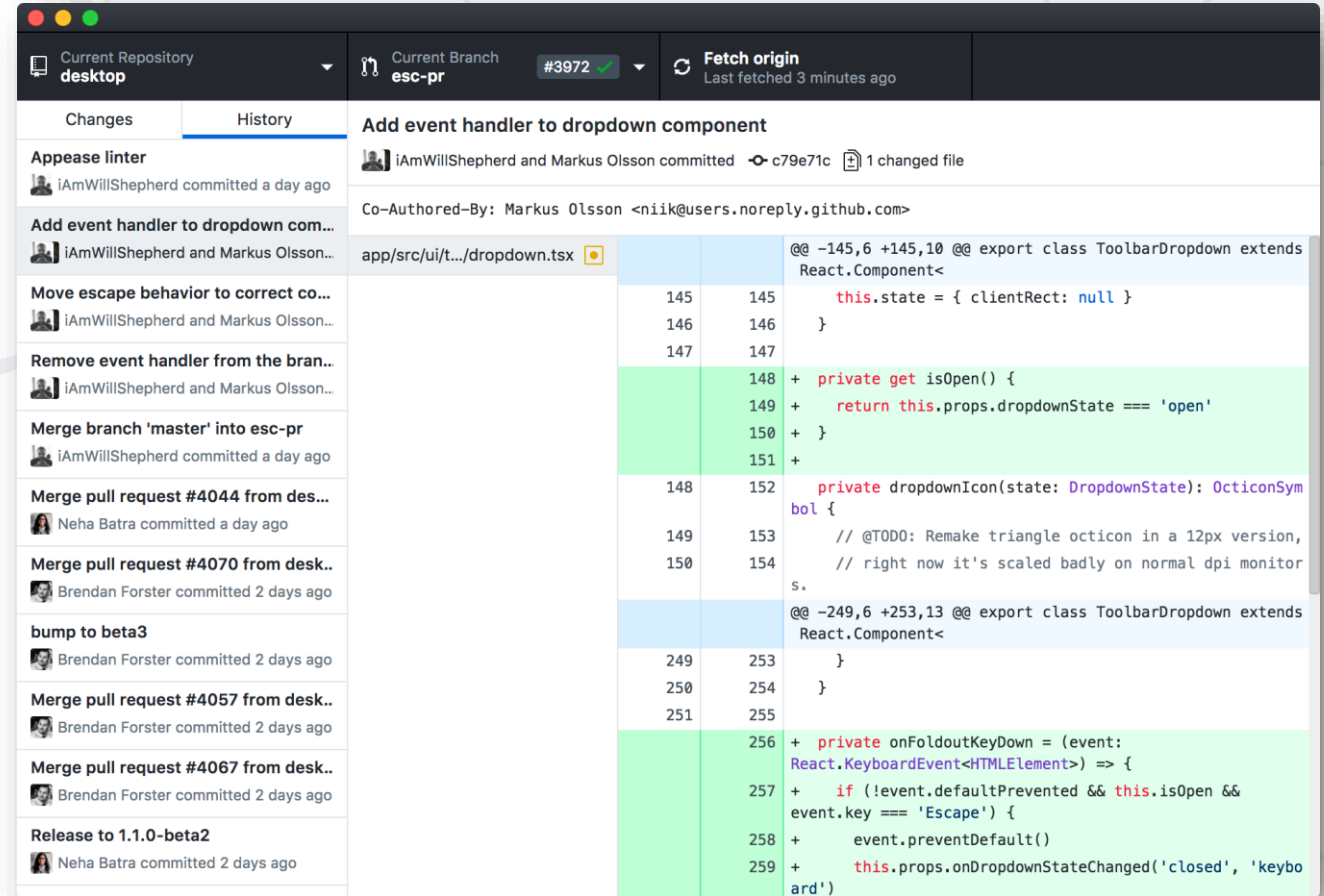


# GitHub Desktop

Development Workflow Simplified

# What is GitHub Desktop?

- Simple **Desktop GUI client** app by GitHub
- Translates Git operations from command-line into a **user-friendly UI**
- Functions: **manage repositories, branches, commits, and pull requests** without opening a terminal



# GitHub Desktop: Key Features

- **Repository** management
  - Create, clone, pull
- **Change** management
  - Add, commit, push
  - Conflict resolution
- **Branch** management
  - Create, switch, merge
  - Pull requests and merging



<https://desktop.github.com>



**GitHub**

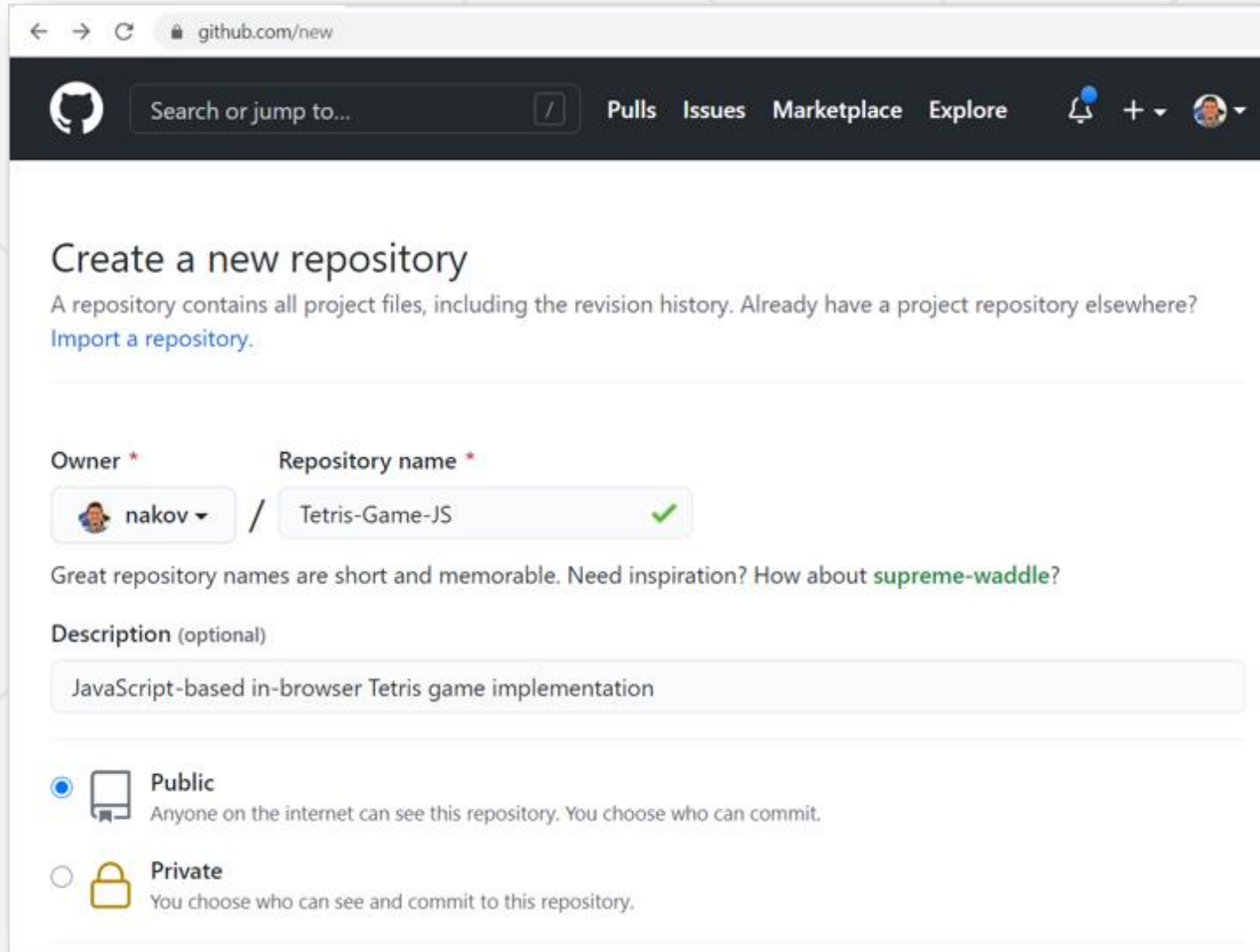
The World's #1 Developer Collaboration Cloud

# What is GitHub?

- **GitHub** is the world's #1 source code hosting site
  - **Free** for open-source projects
  - **Paid plans** for private repositories
- GitHub provides:
  - **Git source code** repository
  - **Issue tracker** (bug tracker)
  - **Project board** (Kanban style)
  - **Wiki pages** (documentation)
  - **Code reviews** (pull requests)
  - **Build system** (actions)
  - **Site hosting** (pages)
  - **Online IDE** (spaces)



# Creating a GitHub Repository




github.com/new

Search or jump to... Pulls Issues Marketplace Explore

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Owner \* Repository name \*


 nakov / Tetris-Game-JS ✓

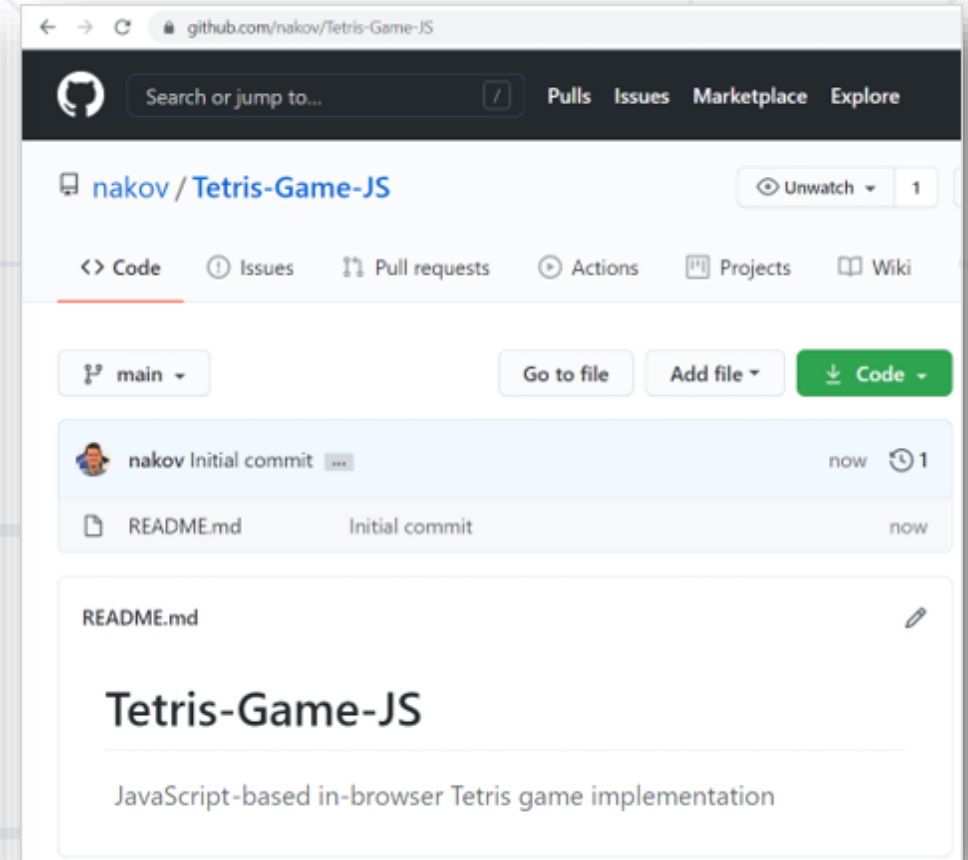
Great repository names are short and memorable. Need inspiration? How about [supreme-waddle?](#)

Description (optional)

JavaScript-based in-browser Tetris game implementation


☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.




github.com/nakov/Tetris-Game-JS

Search or jump to... Pulls Issues Marketplace Explore

 **nakov** / **Tetris-Game-JS** Unwatch 1

<> Code ⓘ Issues 🔗 Pull requests ⚙️ Actions 📁 Projects 📖 Wiki

main Go to file Add file Code

 nakov Initial commit now 1

README.md Initial commit now

README.md

## Tetris-Game-JS

JavaScript-based in-browser Tetris game implementation

- **Clone** a repository from GitHub

```
git clone https://github.com/SoftUni/playground
```

- **Modify** local files

```
notepad README.md
```

- **Commit** changes (local)

```
git add . & git commit -m "Added something"
```

- **Push** the changes to GitHub

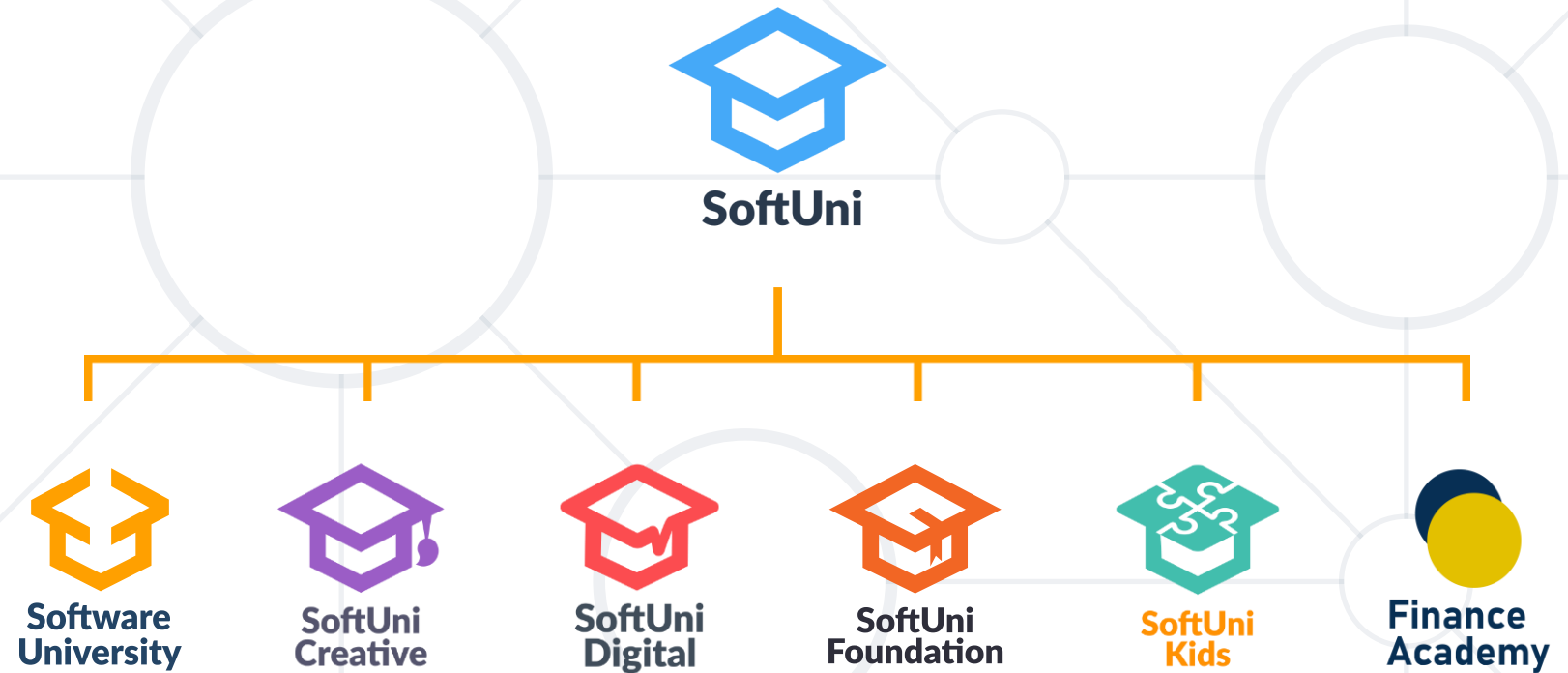
```
git push
```

- Use **version control systems** to work in a team
  - Keep the **shared code** in a central repository
  - Handle merge **conflicts** with ease
- Important **Git** commands:
  - **clone, pull, add, commit, push**
- **Git** clients: **TortoiseGit, GitHub Desktop**
- **GitHub** == the world's most used software project hosting platform
  - Git repository, issue tracker, Kanban board, Wiki





# Questions?



# SoftUni Diamond Partners



- Software University – High-Quality Education, Profession and Job for Software Developers
  - [softuni.bg](http://softuni.bg)
  - Software University Foundation
    - [softuni.foundation](http://softuni.foundation)
- Software University @ Facebook
  - [facebook.com/SoftwareUniversity](https://facebook.com/SoftwareUniversity)
- Software University Forums
  - [forum.softuni.bg](http://forum.softuni.bg)



- This course (slides, examples, demos, exercises, homework, documents, videos and other assets) is **copyrighted content**
- Unauthorized copy, reproduction or use is illegal
- © SoftUni – <https://softuni.org>
- © Software University – <https://softuni.bg>

