

DevOps, DevOps Lifecycle, GitHub, and Diagrams.net

1. DevOps

Introduction What

is DevOps?

DevOps = Development + Operations

It's a **culture + set of practices + tools** that help:

- Developers (who write code)
- Operations (who deploy/manage systems)

→ Work **together**, instead of working in **silos**.

Real-Time Example:

Imagine you're building a mobile app like **Swiggy**.

- The developer writes code to add "Track My Order" feature.
- Without DevOps: Developer gives code to Ops team → they manually deploy → bugs happen → delays.
- With DevOps: Code is **automatically tested and deployed** → customers get updates **faster and safely**.

DevOps Helps You:

- Release features faster (like adding new payment methods in Zomato)
- Fix bugs quickly (like fixing login issue in IRCTC)
- Automate repetitive tasks (like server restarts, database backups)

2. DevOps Lifecycle

The 8 Phases of DevOps Lifecycle:

Phase	Description	Real-Time Example
Plan Mode	Define what to build	Swiggy decides to add a "Dark feature
Develop	Write the code	Developers write Java/Kotlin code for UI
Build	Compile code into deployable format	Android APK is built automatically
Test Mode	Check for bugs automatically	Selenium test checks if "Dark works
Release	Approve the build for deployment	QA signs off the build
Deploy	Push app to servers or app store	APK is uploaded to Play Store
Operate	Keep app running, fix any issues	Monitor if app is crashing
Monitor	Analyze performance and feedback	Firebase or New Relic shows app metrics

| Cycle continues: Feedback from users → back to Plan.

3. GitHub Basics :

What is Git & GitHub?

- **Git** = Version control system (local)
- **GitHub** = Online platform to store Git projects

💡 Real-Life Analogy:

Like **Google Docs** for code. You can:

- Track changes
- Go back to previous versions

- Work together on the same project
-

Real-Time Example:

You're working with a team on an **online shopping website**:

- You add a "Wishlist" page.
- Your friend adds "Product Ratings".
- GitHub helps **merge**, **track**, and **resolve conflicts** if both change same code.

GitHub Hands-On:

☰ Website: **<https://github.com>**

Experiment: 01

Create a GitHub account, then create a new repository with a README file. Upload a basic file named hello.txt, edit the file through GitHub's web interface, commit the changes, and view the commit history. Share the link to your repository after completion.

Step 1: Create a GitHub Account

Go to

<https://github.com>

Click "Sign Up"

Enter:

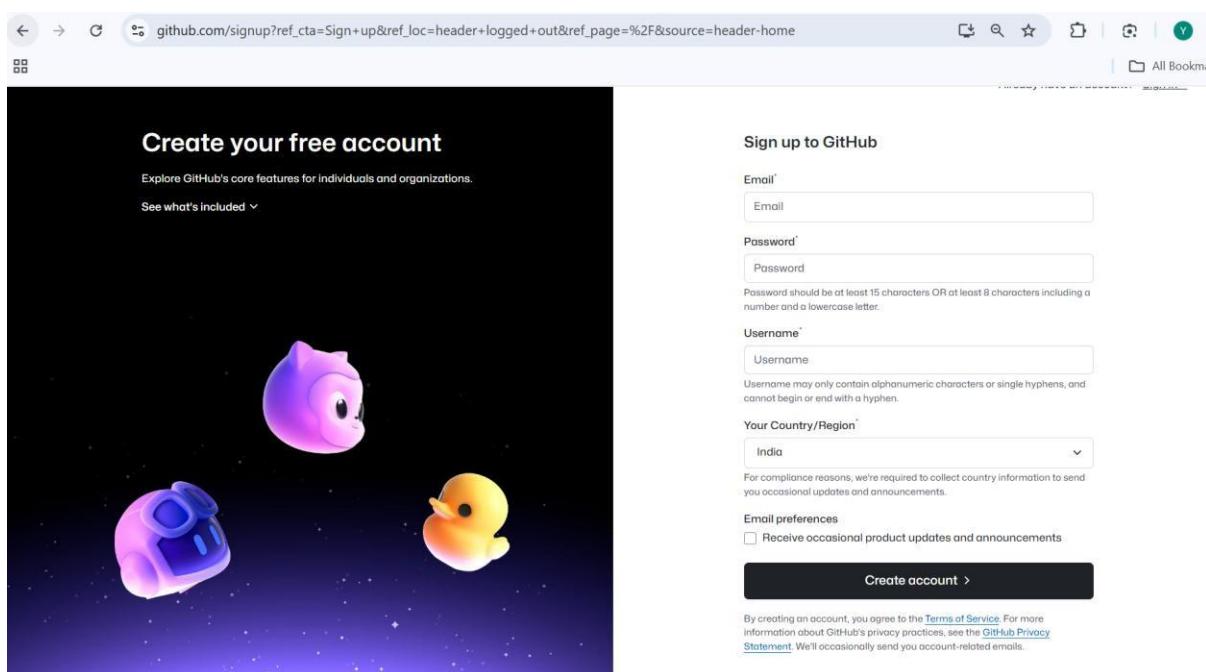
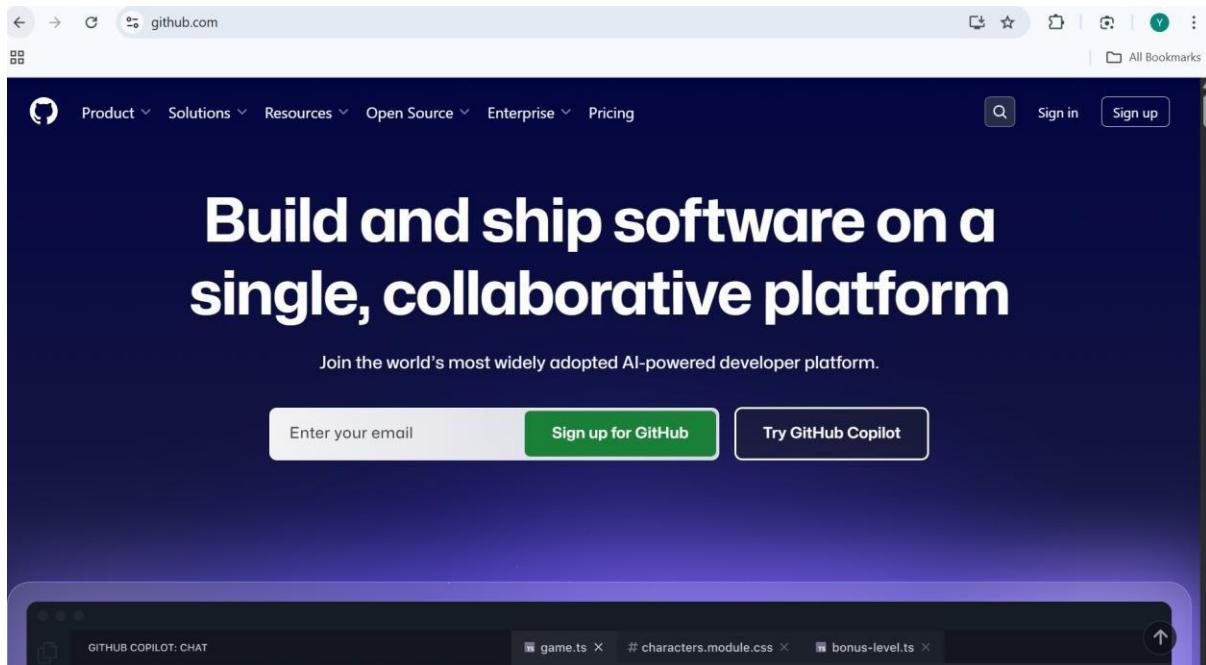
Email

address

Username

Password

Verify your email and log in



Step 2: Create a New Repository

After logging in, click the + icon → New

Repository Fill the form:

Repository name: my-first-repo

Description: Learning GitHub with DevOps

Check: “Initialize this repository with a

README” Click “Create Repository”

The screenshot shows a web browser window with the GitHub homepage loaded. The address bar indicates the URL is `github.com`. The main content area displays the "Home" page with various navigation links like "Ask Copilot", "Get started with GitHub", "Learn to code", and "Create a web app". A prominent feature is a "Playlist" section titled "GitHub for beginners on YouTube" with a thumbnail image of a person speaking. On the right side of the page, there is a vertical sidebar with a dropdown menu open, showing options such as "New issue", "New repository" (which is highlighted with a blue border), "Import repository", "New codespace", "New gist", "New organization", and "New project". Below this sidebar, there is a "Getting started" section with a progress bar labeled "1/3 complete".

At the bottom of the browser window, there is a second tab labeled "New repository" with the URL `github.com/new`. This tab also contains a search bar and other standard browser controls.

The main content area of the browser shows the "Create a new repository" form. The form has two sections: "General" and "Configuration". In the "General" section, the "Owner" dropdown is set to "benjaminDaniel-tech" and the "Repository name" input field contains "my-first-repo". A note below the input field states "my-first-repo is available.". There is also a suggestion "Great repository names are short and memorable. How about didactic-octo-disco?". The "Description" field contains the text "Learning GitHub with DevOps". In the "Configuration" section, the "Choose visibility" dropdown is set to "Public".

2 Configuration

Choose visibility *
Choose who can see and commit to this repository

Public

Add README
READMEs can be used as longer descriptions. [About READMEs](#)

On

Add .gitignore
.gitignore tells git which files not to track. [About ignoring files](#)

No .gitignore

Add license
Licenses explain how others can use your code. [About licenses](#)

No license

Create repository

The screenshot shows a GitHub repository page for 'my-first-repo'. The repository is owned by 'benjaminDaniel-tech'. The main navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The repository name 'my-first-repo' is displayed prominently, along with a 'Public' badge. Below the repository name, there are stats: 1 Branch, 0 Tags, and 1 Commit. A commit history shows an 'Initial commit' by 'benjaminDaniel-tech' with the commit hash ee4b303. The repository contains a single file, 'README.md', which is shown as a rich text editor with the title 'my-first-repo' and the subtitle 'Learning GitHub with DevOps'. The right sidebar provides options for learning, reading, and creating releases.

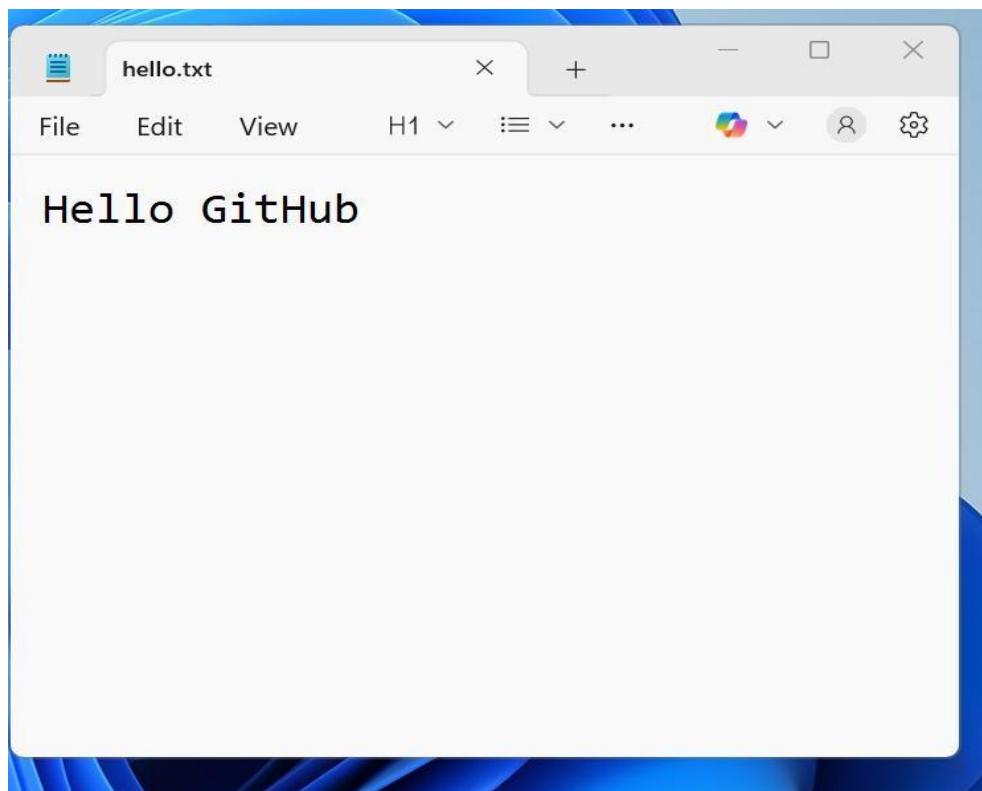
Step 3: Upload a File (hello.txt)

Inside your new repo, click “Add file” → “Upload files” Drag & drop or select a file named hello.txt

You can create it using Notepad or just type “Hello GitHub” inside Scroll down → Add a commit message: **added hello.txt**

Click “Commit changes”

The screenshot shows a GitHub repository named "my-first-repo". The repository is public and has one branch ("main") and no tags. A commit from "benjaminDaniel-tech" is visible, titled "Initial commit" and containing the file "README.md". The README file content is "Learning GitHub with DevOps". A context menu is open over the "README" file, showing options to "Create new file" and "Upload files".



my-first-repo /



Drag files here to add them to your repository

Or choose your files

↑ [] > Daniel - Personal > Desktop v C Search Desktop

New folder

Hello GitHub

File	Type	Size
hello	Text Document	12 bytes
Day 1 - Devops Introduction	Chrome HTML Document	3.80 MB
Git tool	Chrome HTML Document	5.09 MB
Selecting Git for Windows	Microsoft Word Document	2.87 MB
VMRDA Invoice	Microsoft Word Document	56.4 KB
2	Microsoft Word Document	1.69 MB
hdfc	Chrome HTML Document	139 KB
HDFC Letter	Chrome HTML Document	145 KB
1_merged_17_merged_merge d-numbered (1)	Chrome HTML Document	
Resume - Daniel	Chrome HTML Document	
Client Name_MIQ_Razorpay On-boarding	Microsoft Word Document	

File name: All Files

Open

Upload files · benjaminDaniel-tech · +

github.com/benjaminDaniel-tech/my-first-repo/upload/main

benjaminDaniel-tech / my-first-repo

Type / to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

my-first-repo /



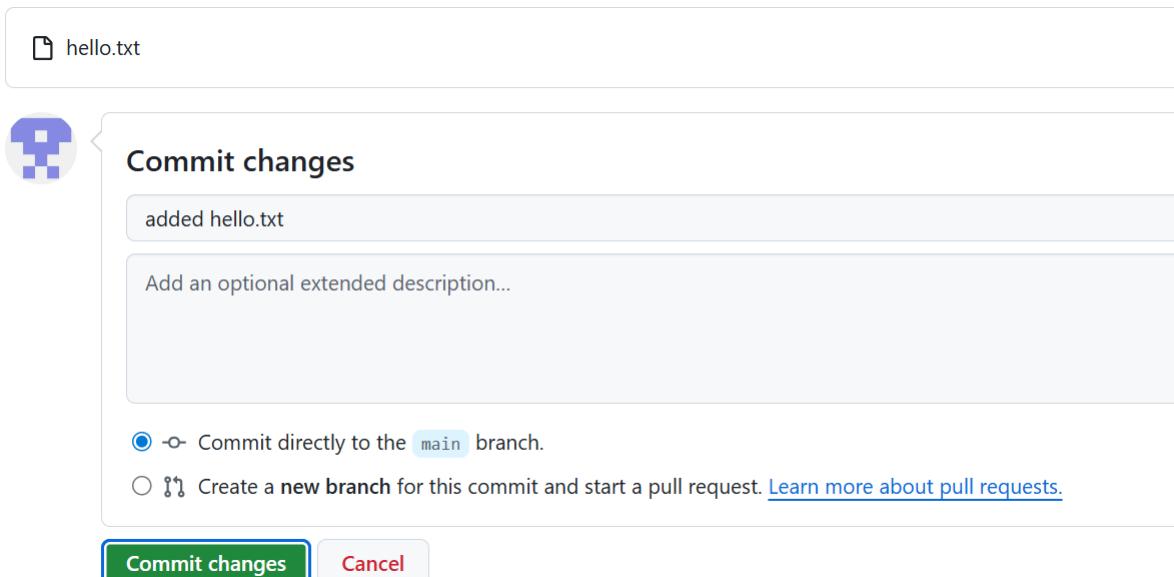
Drag additional files here to add them to your repository

Or choose your files

hello.txt



Commit changes



The screenshot shows the GitHub repository page for "my-first-repo". The repository is public and has 2 commits. The commits are:

- benjaminDaniel-tech added hello.txt (9d20c2d · now)
- Initial commit (6 minutes ago)

The repository page also includes sections for README, Activity, Stars, Forks, Releases, and Packages.

Step 4: Edit the File Online

Click on hello.txt from the repo file

list Click the pencil icon (Edit this file)

Change the content, e.g., add your name or a

second line Scroll down, add a commit message:

updated hello.txt

Click Commit changes

my-first-repo Public

main 1 Branch 0 Tags

Go to file Add file Code

benjaminDaniel-tech added hello.txt 9d20c2d · 1 minute ago 2 Commits

README.md Initial commit 7 minutes ago

hello.txt added hello.txt 1 minute ago

my-first-repo / hello.txt

benjaminDaniel-tech added hello.txt 9d20c2d · 2 minutes ago History

Code Blame 1 lines (1 loc) · 12 Bytes

Raw Edit this file

```
1 Hello GitHub
```

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Files my-first-repo / hello.txt in main

Cancel changes Commit changes...

main Go to file README.md hello.txt

```
1 Hello GitHub,  
2  
3 My Name is Daniel Benjamin!
```

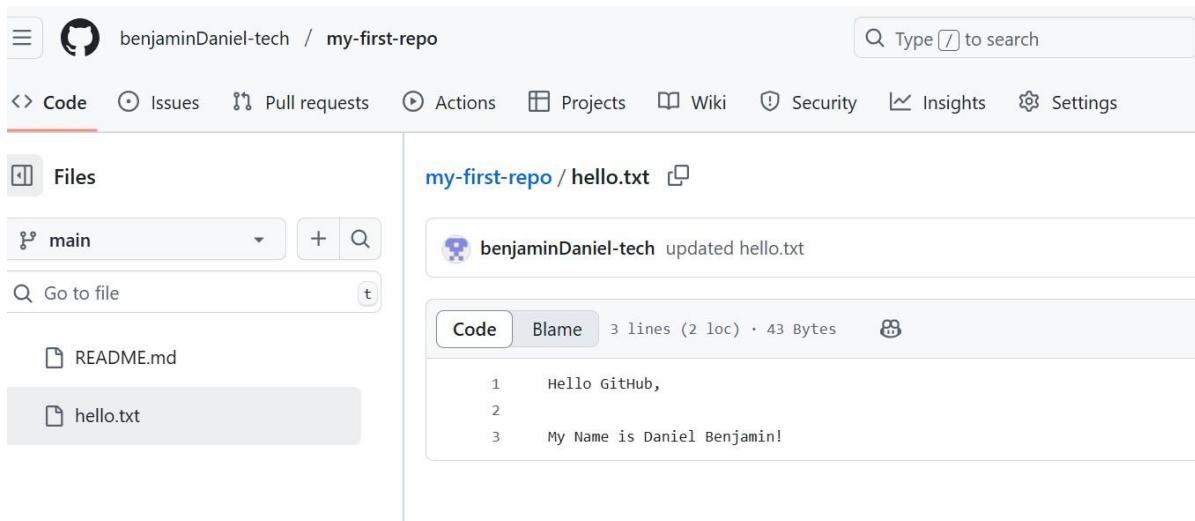
Commit changes

Commit message updated hello.txt

Extended description Add an optional extended description...

Commit directly to the main branch
 Create a new branch for this commit and start a pull request [Learn more about pull requests](#)

Cancel Commit changes



A screenshot of a GitHub repository page. The top navigation bar shows the repository name 'benjaminDaniel-tech / my-first-repo'. The 'Code' tab is selected. The main content area shows the repository details: 'my-first-repo' (Public), 1 Branch, 0 Tags. It lists two commits: 'benjaminDaniel-tech updated hello.txt' (1 minute ago) and 'Initial commit' (18 minutes ago). Below the commits is a section for 'README' with the text 'my-first-repo' and a note 'Learning GitHub with DevOps'.

Step 5: View Version History

At the top of your repo, click the “Commits” tab

You will see all your commit messages and time of changes Click on a commit to view the file version at that point

012 · 5 minutes ago 2 Commits

Commits

main

Commits on Aug 3, 2025

- updated hello.txt**
benjaminDaniel-tech authored 5 minutes ago 17de012
- added hello.txt**
benjaminDaniel-tech authored 17 minutes ago 9d20c2d
- Initial commit**
benjaminDaniel-tech authored 23 minutes ago ee4b303

Commit 17de012

benjaminDaniel-tech authored 6 minutes ago

updated hello.txt

main

1 parent 9d20c2d commit 17de012

Filter files...

1 file changed +3 -1 lines changed

hello.txt

```

@@ -1 +1,3 @@
1 - Hello GitHub
1 + Hello GitHub,
2 +
3 + My Name is Daniel Benjamin!

```

Comments 0

Search within code

Customizable line height

The default line height has been increased for improved accessibility. You can choose to enable a more compact line height from the view settings menu.

Enable compact line height Dismiss

Final Submission:

Copy your repo link (example: <https://github.com/yourname/my-first-repo>) Example: <https://github.com/benjaminDaniel->

[tech/my-first-repo](https://github.com/tech/my-first-repo)

Paste it into your classroom form / send it to trainer

Experiment: 02

Create a new file named about.txt in your GitHub repository. Inside the file, write your name, course, and one thing you learned about GitHub. Commit the file to your repository and share the link after completing the task.

Experiment: 03

Create a new GitHub repository named student-portal-demo. Initialize it with a README file describing a basic idea for a student portal application. Share the link to your repository after completing the task.

DevOps Visualizing: Bringing Processes to Life with diagrams.net

- ➔ DevOps involves multiple stages — like Plan, Develop, Build, Test, Release, Deploy, Operate, Monitor.
- ➔ Using diagrams.net helps students see the full flow clearly, instead of just reading theory.

“If you can visualize it, you can understand it better.”

Common Shapes & Their Meanings:

Shape	Name/Type	Meaning / Use Case
-------	-----------	--------------------

 Rectangle	Process / Task	Represents a step in a process (e.g., <i>Build App, Compile Code</i>)
 Diamond	Decision	Used for yes/no or true/false conditions (e.g., <i>Build successful?</i>)
 Oval / Ellipse	Start / End	Represents start or end of a process or flow
 Parallelogram	Input / Output	Denotes input/output operations (e.g., <i>Get user input, Display results</i>)
 Document Shape	Document	Represents a file, report, or documentation
 Cylinder	Database / Storage	Used for databases or storage (e.g., <i>MySQL DB, Blob Storage</i>)
 Folder Shape	Data Set / Repository	Used to represent a collection or a repository (e.g., <i>GitHub Repo</i>)
 Cloud Shape	Cloud / Internet	Represents cloud services or external internet systems (e.g., <i>AWS, Azure</i>)
 Arrows	Connector	Show the direction or flow between steps or components

Shape	Name/Type	Meaning / Use Case
Cube / 3D Box	Service / Microservice	Represents a standalone service (e.g., <i>Authentication Service</i>)
 Gear Shape	Operation / Automation Tool	Indicates automated tools or processes (e.g., <i>CI/CD via Jenkins</i>)

Person Shape	Actor / User	Represents a human user or external system interacting with the process
Circle (labeled)	Connector Node	For off-page connectors or complex flows (help break big diagrams into sections)
Note / Sticky	Annotation / Comments	Used to add explanations, notes, or references in diagrams

You can **find these shapes** in the left sidebar of diagrams.net under sections like:

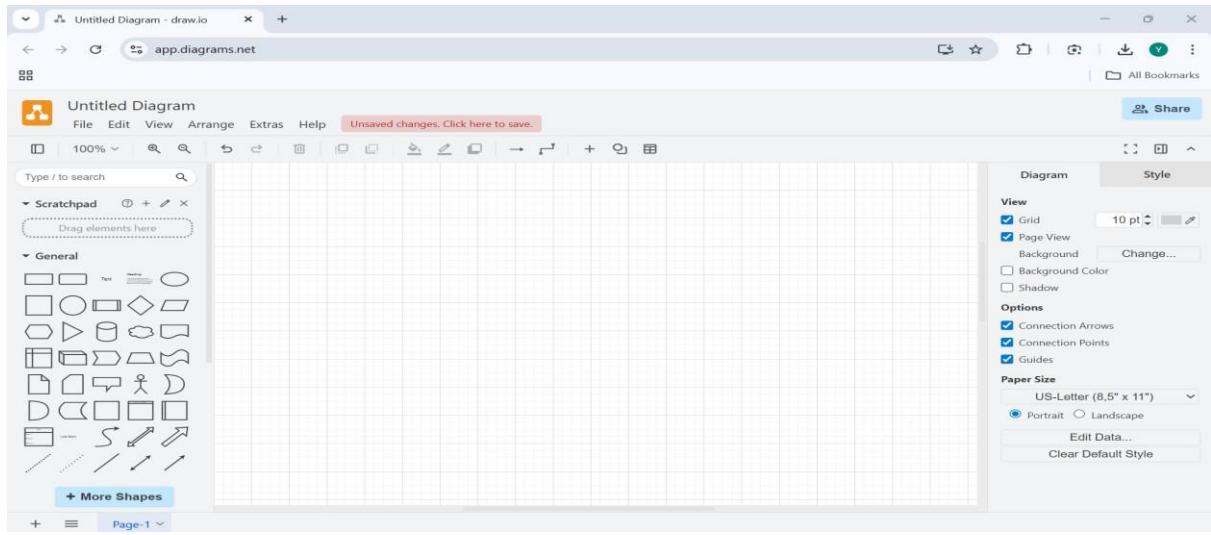
- General
- Flowchart
- AWS / Azure / UML / Network (advanced users)

Experiment: 04

Design a DevOps Lifecycle Using **diagrams.net Tool:**

<https://app.diagrams.net> (also called Draw.io)

No login required. Select “Device” when it asks where to save.



Objective:

Visually represent the **DevOps Lifecycle**, using basic shapes and arrows to show how software moves from **Planning to Monitoring**.

Steps to Follow:

Open diagrams.net

Visit <https://app.diagrams.net>

Choose Storage Location

Select “Device” when prompted and click Create New

Diagram Set File Name

Name it: devops-lifecycle

Drag and Drop Shapes to Build the Lifecycle

Use rounded rectangles for each stage:

Plan → Develop → Build → Test → Release → Deploy → Operate →

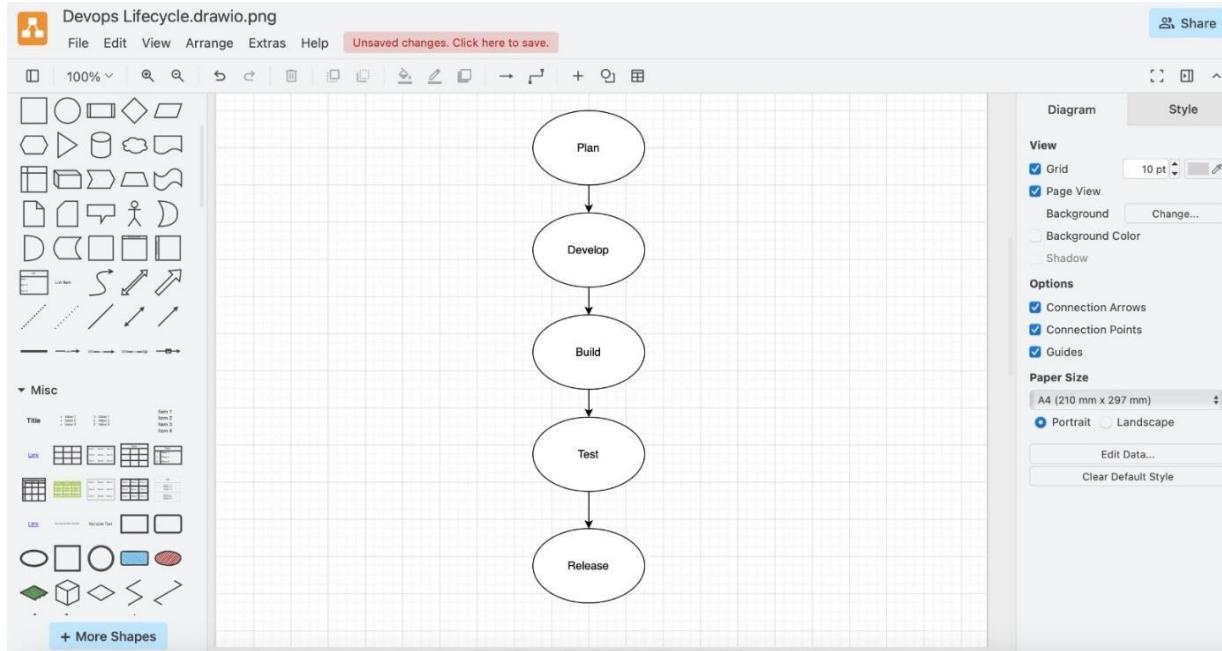
Monitor Use arrows to connect them in a circular or linear format

Add a Title at the Top:

Example: “DevOps Lifecycle – Visual Diagram”

Export Diagram as PNG:

File → Export As → PNG → Click “Download” as shown below..



DevOps Lifecycle with Shapes & Usage:

DevOps Stage	Recommended Shape	Meaning / Purpose
Plan	■ Rectangle	Define project goals, gather requirements, and prepare backlog
Develop	■ Folder / ■ Rectangle	Write and manage source code in a version control system
Build	⚙ Gear / ■ Rectangle	Compile source code, create build artifacts
Test	◆ Diamond / ■ Rectangle	Run automated/manual tests to validate the build
Release	■ Rectangle / ○ Oval	Approve and package code for deployment
Deploy	☁ Cloud / ■ Cylinder	Deploy code to staging or production environments (live servers or in Play store)

Operate	Gear / Rectangle	Maintain and manage the application in the live environment
----------------	------------------	-------------------------------------------------------------

DevOps Stage	Recommended Shape	Meaning / Purpose
Monitor	Parallelogram / Cloud	Track performance, collect logs, and gather feedback

Upload to GitHub:

Go to your repository

Click “Add file → Upload files”

Upload the exported PNG file

Commit with message: **Added DevOps Lifecycle diagram**

The screenshot shows a GitHub repository interface. At the top, there's a header with a user icon, the repository name 'benjaminDaniel-tech/My-first-devops', and a search bar. Below the header, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The 'Code' link is underlined, indicating it's the active tab. The main content area displays the repository details: 'main' branch, 1 Branch, 0 Tags. There are two commits listed:

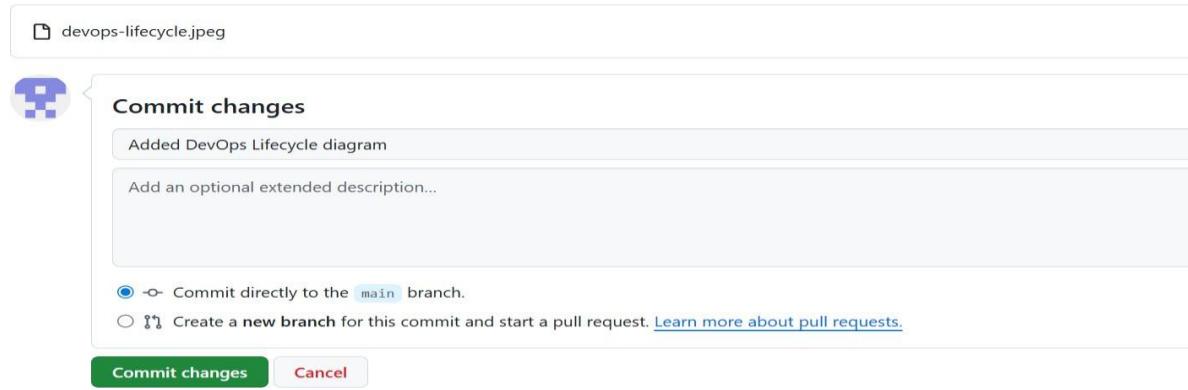
- benjaminDaniel-tech Add files via upload · 9fa45e5 · 5 hours ago
- Initial commit
- demo.txt Add files via upload · 5 hours ago

At the bottom of the repository page, there's a section for the 'README' file, which is currently being edited, as indicated by the 'Edit file' button.

A screenshot of a GitHub repository page titled "My-first-devops". The repository is public and has 1 branch and 0 tags. The main file listed is "README.md", which was added via upload by "benjaminDaniel-tech" as an initial commit. Below it is a file named "demo.txt" also added via upload. A context menu is open over the "demo.txt" entry, showing options like "Create new file" and "Upload files". The "Code" tab is selected at the top.

A screenshot of a GitHub file upload interface titled "Upload files · benjaminDaniel-tech". It shows the URL "github.com/benjaminDaniel-tech/My-first-devops/upload/main". The repository navigation bar is visible at the top, with "Code" being the active tab. A large text area in the center says "Drag files here to add them to your repository" or "Or choose your files".

A screenshot of a GitHub file upload interface titled "My-first-devops /". It shows the URL "github.com/benjaminDaniel-tech/My-first-devops". The repository navigation bar is visible at the top, with "Code" being the active tab. A large text area in the center says "Drag additional files here to add them to your repository" or "Or choose your files". Below this area, a file named "devops-lifecycle.jpeg" is listed with a small thumbnail icon.



The screenshot shows the GitHub repository page for 'My-first-devops'. The repository is public and contains 3 commits. The commits are:

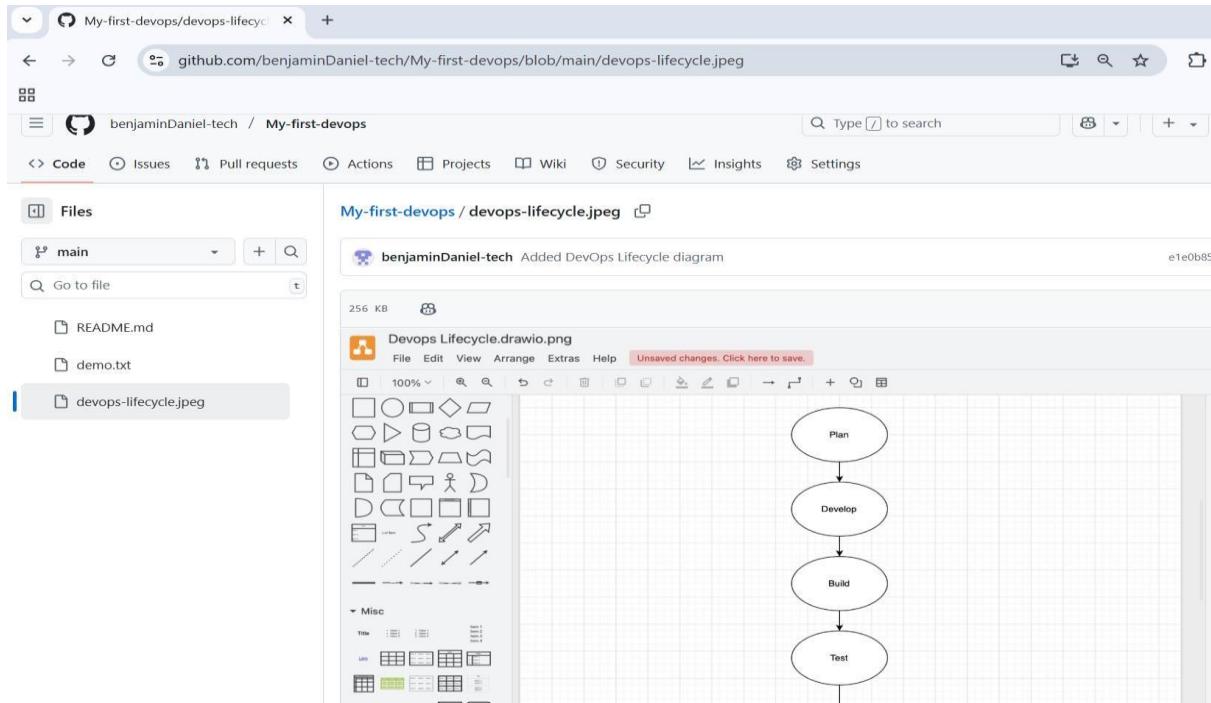
File	Message	Time
benjaminDaniel-tech/README.md	Initial commit	5 hours ago
benjaminDaniel-tech/demo.txt	Add files via upload	5 hours ago
benjaminDaniel-tech/devops-lifecycle.jpeg	Added DevOps Lifecycle diagram	now

The repository page also shows a 'Code' tab is selected, and there is a 'README' file listed. On the right side, there are various repository statistics and settings.

Expected File Name in GitHub Repo:

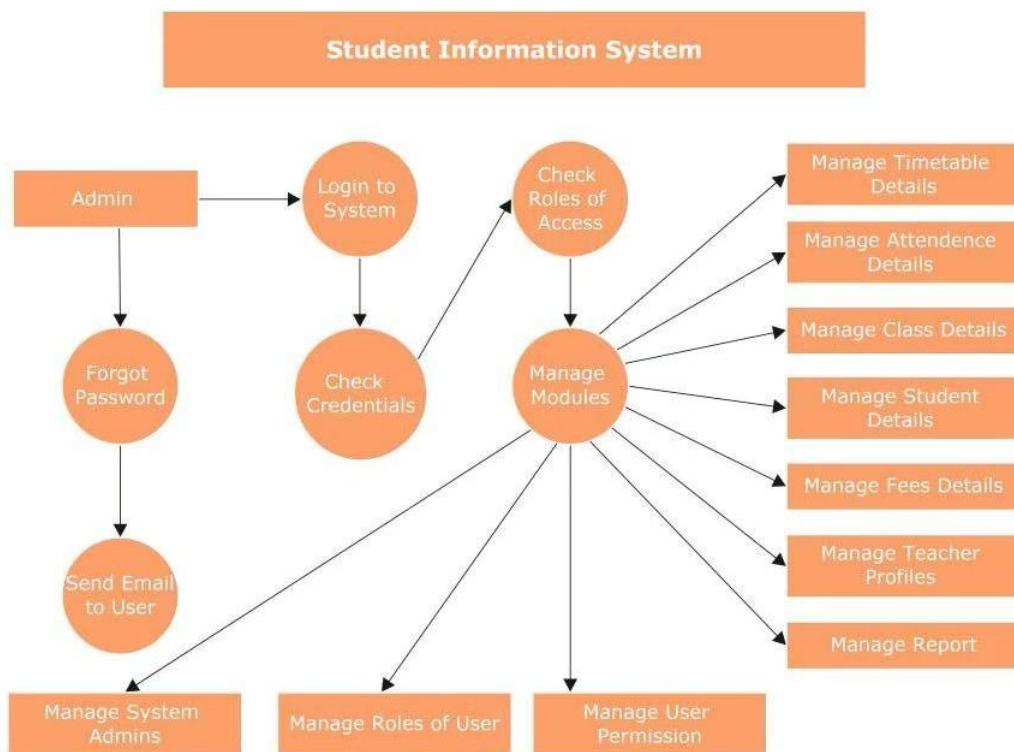
devops-lifecycle.png

shown below..



Experiment: 05

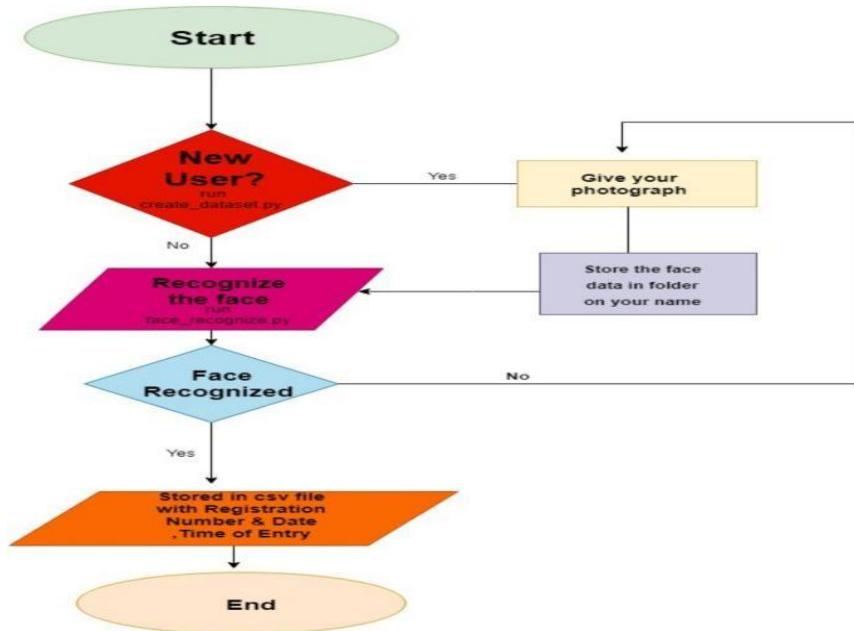
Design a DevOps Lifecycle diagram for a Student Information System.



and make a note of the URL for future reference.

Experiment: 06

Design a DevOps Visualization Diagram – Face Recognition Attendance System



and make a note of the URL for future reference.