



EXPERIENTIAL LEARNING & GLOBAL ENGAGEMENT

OPEN SOURCE ENGINEERING

Student ID: 2400031752

Name: M. DHARANI REDDY

KL University

1 About Linux Distro You Have Used

1.1 1. Overview of the Distribution

The Linux distribution I used for this course is **Ubuntu 24.04.3 LTS**. It is a free, open-source operating system built on the Debian base and maintained by Canonical. Ubuntu is widely known for being stable, secure, and easy to use, which makes it one of the most beginner-friendly Linux distros. Its LTS (Long-Term Support) version ensures reliable updates and long-term stability for everyday tasks and development activities.

1.2 2. Ubuntu's Philosophy and Design

Ubuntu follows the principle “**Linux for human beings**,” focusing on accessibility and user comfort. It is designed to give a smooth experience to first-time users while still being powerful enough for programmers and professionals. Its emphasis on simplicity, security, and usability makes it suitable for learning, productivity, and open-source development.

1.3 3. Desktop Environment (GNOME)

Ubuntu uses the **GNOME** desktop environment, which provides a clean and modern graphical interface. It features:

- A left-side dock for quick access
- An Activities Overview for multitasking
- A simple, distraction-free layout
- Easy window and workspace management

The interface supports efficient navigation and feels similar to other modern operating systems, making it comfortable for new users.

2 Encryption and GPG

2.1 Definition

Encryption is a security technique used to convert readable data into an unreadable or scrambled format. This protects information from unauthorized access. Only a person who has the correct **key** can convert the encrypted data back to its original, readable form. Encryption plays a major role in ensuring:

- **Privacy** – keeping information confidential
- **Security** – protecting sensitive data from hackers
- **Integrity** – preventing unauthorized modifications
- **Safe communication** – especially over the internet

Encryption is widely used in email communication, banking systems, cloud storage, messaging apps, and secure file sharing.

2.2 GPG (GNU Privacy Guard) Explained

GPG is the GNU implementation of the **OpenPGP** standard (originally Pretty Good Privacy - PGP). It is essential for protecting individual files and ensuring secure, authenticated communication.

2.2.1 Core GPG Concepts

GPG relies on **asymmetric cryptography**, which uses a pair of mathematically linked keys:

- **Public Key:** This key is shared with everyone. It can be used to **encrypt** a message that only you can read, or to **verify** a signature you created.
- **Private (Secret) Key:** This key is kept **secret** and is protected by a strong passphrase. It is used to **decrypt** messages sent to you, or to **digitally sign** files to prove they came from you.

2.2.2 Basic GPG Command-Line Usage

GPG is usually pre-installed on Ubuntu and is primarily used through the command line (Terminal).

2.3 Basic GPG Commands (Short and Expanded Version)

A. Generating a Key Pair The first step in using GPG is creating your own key pair. This generates a **public key** (to share with others) and a **private key** (kept secret):

```
gpg --full-generate-key
```

You will be asked to choose the key type, key size, your name, email, and a strong passphrase to secure your private key.

B. Encrypting a File (Symmetric Encryption) Symmetric encryption uses a single passphrase to protect your file. It is quick and suitable for personal files:

```
gpg -c myfile.txt
```

This creates an encrypted version called `myfile.txt.gpg`, which can only be opened by entering your passphrase.

C. Encrypting for Someone Else (Asymmetric Encryption) This method is used when sending secure files to another person. You must have their **public key** imported into your system:

```
gpg --encrypt --recipient "email@example.com" file.doc
```

Only the intended recipient, who has the matching private key, can decrypt and open the file.

D. Decrypting a File To open an encrypted file sent to you, use:

```
gpg --decrypt file.doc.gpg
```

GPG will ask for the passphrase of your private key before revealing the original file. You may add `--output` to save the decrypted file to a specific location.

3 Sending Encrypted Email

Sending encrypted email is a secure method of communication that ensures only the intended recipient can read the message. Ubuntu commonly uses **GPG (GNU Privacy Guard)** for this purpose, which relies on public-key cryptography. The sender encrypts the email using the recipient's **public key**, and the recipient decrypts it using their **private key**.

The process involves several steps:

- **Public Key Exchange:** Both users generate a GPG key pair and share their public keys with each other. Public keys are safe to share and can even be uploaded to key servers.
- **Importing the Recipient's Key:** Before sending an encrypted email, the sender must import the recipient's public key into their keyring.
- **Encrypting the Message:** Once the key is imported, the email content or attached file is encrypted using GPG. The encrypted text appears unreadable to anyone without the corresponding private key.
- **Sending the Email:** The encrypted message can be pasted into an email client or sent as an encrypted attachment.
- **Decryption:** The recipient uses their private key to decrypt the email and restore it to its original readable form.

Encrypted email ensures **confidentiality**, **authentication** through digital signatures, and **integrity** of the message. It protects sensitive communication from unauthorized access, interception, and tampering during transmission.

4 Privacy Tools from PRISM-Break

Below are five recommended privacy-focused tools listed on prism-break.org. These tools help protect user data, communication, and online activity.

4.1 1. Tor Browser

Tor Browser allows anonymous web browsing by routing internet traffic through multiple encrypted nodes. It hides the user's IP address and prevents websites or trackers from monitoring activity.

4.2 2. KeePassXC

KeePassXC is a secure, open-source password manager that stores all passwords in an encrypted local database. It avoids cloud storage and ensures complete control over sensitive credentials.

4.3 3. Signal

Signal is an end-to-end encrypted messaging application used for private chats, voice calls, and video calls. It does not store metadata and prioritizes user privacy.

4.4 4. Syncthing

Syncthing is a peer-to-peer file synchronization tool that allows devices to share files directly without using third-party cloud servers. All transfers are encrypted and decentralized.

4.5 5. ProtonMail

ProtonMail is an encrypted email service based in Switzerland. It provides end-to-end encryption, meaning even the service provider cannot read user emails.

5 Open Source License

Certainly. Here is the information about the **MIT License** organized into clear, descriptive headings, strictly maintaining a paragraph-only format within each section.

5.1 The Core Purpose and Classification

The MIT License is renowned as one of the most permissive and concise open-source licenses currently in use. Originating from the Massachusetts Institute of Technology, its primary goal is to encourage maximum adoption and reuse of software with minimal legal friction. It is formally classified as a **permissive license**, meaning it grants users broad rights to use, modify, and distribute the software without imposing the reciprocal sharing obligations seen in copyleft licenses, such as the GNU General Public License (GPL). This makes the MIT License highly favorable for both commercial enterprises and proprietary software development.

5.2 Granted Rights and Permissions

The license grants blanket permission to any individual or entity obtaining a copy of the software and its associated documentation to deal with the Software without restriction. Specifically, users are granted explicit rights to **use, copy, modify, merge, publish, distribute, sublicense, and/or sell** copies of the software. This expansive grant allows developers to incorporate MIT-licensed code into projects that may ultimately be closed-source and sold commercially, provided they meet the few mandated conditions.

5.3 The Only Two Conditions for Distribution

Unlike licenses that enforce reciprocal sharing, the MIT License has only two critical requirements that must be met when the software is distributed or included in a larger work. The first condition is the mandatory inclusion of the original **Copyright Notice** (e.g., Copyright <YEAR> <COPYRIGHT HOLDER>). The second is the mandatory inclusion of the full **License Text** itself. If these two simple requirements are satisfied, the user can otherwise treat the code as they wish, including releasing their modifications under a proprietary license.

5.4 Installation Process

6 Self Hosted Server

6.1 About

FlatPress is a simple, lightweight, and database-free content management system designed mainly for blogging and small websites. What makes FlatPress unique is that it uses **flat files** to store all data instead of relying on traditional databases like MySQL. This design choice makes it extremely easy to install, maintain, and move between servers. Since it only requires PHP, FlatPress can run on almost any hosting environment, even very low-cost or limited ones. This simplicity also reduces the chances of database-related security risks and errors, making FlatPress a stable option for users who want a minimal system without too much technical complexity.

FlatPress offers all the essential features expected from a modern blogging platform. Users can create posts and pages directly from a clean and simple admin panel. Themes allow customization of the site's appearance, and plugins can extend the functionality as needed, such as adding contact forms, enhancing SEO, or improving the editor experience. The built-in comment system allows readers to interact, and additional plugins can add moderation or spam protection features. Because all content is stored in text files, backups are extremely easy—copying the site directory is usually enough to secure everything. This also makes FlatPress highly portable: you can move the entire site simply by uploading the folder to another server.

6.2 Installation Process (Flatpress)

To install FlatPress, first ensure your server supports PHP 5.2 or higher and has a web server such as Apache or Nginx. Unlike many CMS platforms, FlatPress does not require a database, which makes it lightweight and easy to set up. Begin by downloading the latest version of FlatPress from the official website and extracting the ZIP file on your computer. The extracted folder contains key directories like `fp-admin` for the admin panel, `fp-content` for themes, plugins, posts, and pages, as well as files like `index.php` and `fp-config.php` for configuration. Upload all these files to your web server using an FTP client or your hosting file manager, either to the root directory or a subfolder. After uploading, open the `fp-config.php` file in a text editor to configure your site settings, including the site title, admin username and password, and admin email. Make sure the `fp-content` folder and its subdirectories are writable so that FlatPress can store content. Once uploaded and configured, access the admin panel by navigating to `http://yourdomain.com/fp-admin/` in your browser and log in using your admin credentials. From the admin panel, you can create posts and pages, install themes and plugins, and manage comments. Because FlatPress stores all content in plain text files, it is highly portable and easy to back up—simply copy the folder to another server to migrate the site.

Flat Press



7 Open Source Contribution

7.1 PR 1 : First Contribution

7.1.1 Goal

The project's objective is to simplify the standard open-source contribution workflow, allowing beginners to easily add their name to the project's `Contributors.md` file.

7.1.2 The Contribution Workflow

The tutorial details the standard **fork - clone - edit - pull request** sequence, essential for collaborative coding.

7.1.3 1. Setup

- **Fork:** Create a copy of the repository in your personal GitHub account.
- **Clone:** Download the forked repository to your local machine using the `git clone` command and the SSH URL.
- **Prerequisites:** Ensure **Git** is installed; alternatives for users uncomfortable with the command line (GUI tools) are provided.

7.1.4 2. Making Changes

- **Branch:** Create a new isolated branch for your changes using `git switch -c your-new-branch-name`.
- **Edit:** Add your name to the `Contributors.md` file using a text editor.

- **Commit:** Stage the changes with `git add Contributors.md` and save them locally with `git commit -m "Add your-name to Contributors list".`

7.1.5 3. Submission

- **Push:** Upload your local branch to your GitHub fork using `git push -u origin your-branch-name`.
- **Pull Request (PR):** Go to your GitHub repository and submit a PR via the "Compare & pull request" button for review by the project maintainers.

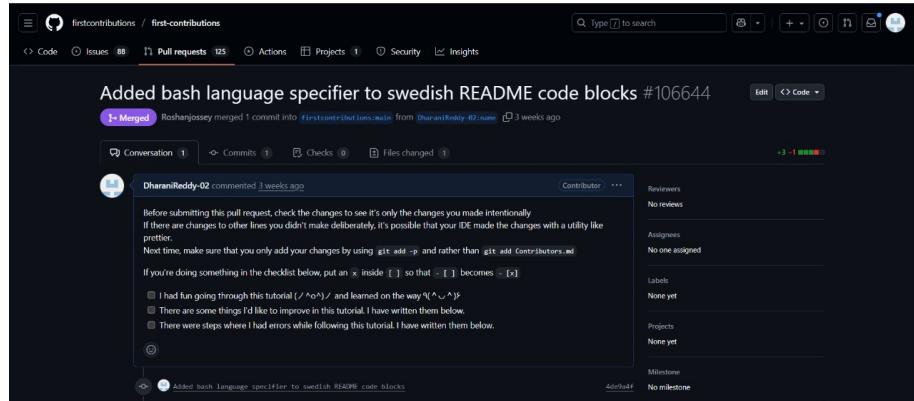
7.1.6 Difficulties and Solutions

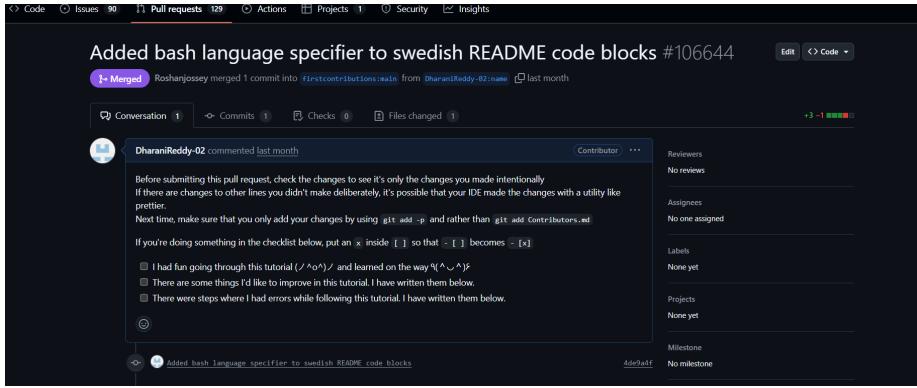
The guide anticipates and solves two common beginner issues:

- **Old Git Version:** If the `git switch` command fails, use the older command: `git checkout -b your-new-branch`.
- **Authentication Error:** If `git push` fails due to GitHub removing password support, the solution is to configure an **SSH key** or a **Personal Access Token** and ensure your remote URL is set to the **SSH protocol** (`git remote set-url origin git@github.com:....`).

7.1.7 Next Steps

Upon merging the PR, the user is encouraged to celebrate their first contribution and seek out other beginner-friendly issues on the project list.





7.2 PR 2

The *javascript-algorithms* repository is a widely used open-source project that provides clean, well-structured implementations of classic data structures and algorithms using JavaScript. It is designed as an educational resource, helping learners understand how algorithms work internally through readable code, examples, and test cases. The project functions like a complete learning reference for data structure implementations, supporting students, developers, and contributors who want to understand algorithmic logic beyond library-based usage. It also serves as a highly active community-driven platform where contributors can improve existing implementations, fix issues, or add missing features, making it an evolving and collaborative learning space.

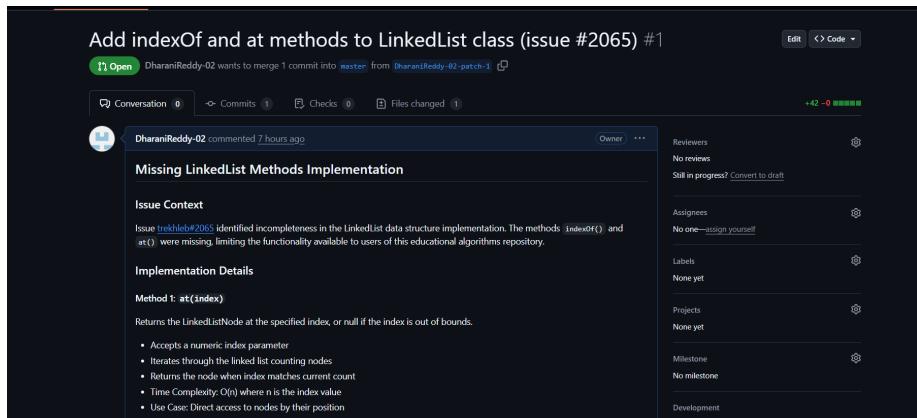
7.2.1 Licensing and Self-Hosting Options

In this repository, my pull request addressed an incompleteness in the existing **LinkedList** data structure implementation. Issue #2065 had identified that two essential methods—`at()` and `indexOf()`—were missing, which limited the ease of accessing nodes by position and searching for elements within the list. To improve functionality and support learners, I implemented both methods, ensuring they align with the repository's coding style, logic flow, and data structure behavior.

The `at()` method enables users to access a node at a specific index. It sequentially traverses the linked list starting from the head, counting nodes until it reaches the requested index. If the index is valid, it returns the corresponding node; otherwise, it returns `null`. This method provides predictable and intuitive node access similar to array indexing, while still reflecting the inherent linear traversal nature of linked lists.

7.2.2 Community and Support

The *javascript-algorithms* project maintains an active and collaborative open-source community centered around GitHub. Contributors regularly open issues, propose improvements, and submit pull requests to enhance algorithm implementations or fix inconsistencies. Discussions often happen directly within GitHub issues and PR threads, allowing contributors to receive feedback, share ideas, and refine their work. The repository includes extensive documentation, examples, and test cases that help newcomers understand how to contribute effectively.



7.3 PR 3 : Y24 Open Source Engineering

7.3.1 Introduction and Purpose

FlatPress is a **lightweight, open-source blogging platform** that works **without any database**. Instead of MySQL or MongoDB, it stores everything (posts, settings, plugins, themes) in simple **text files**. This makes it **very easy to install, move, and back up**.

```

...
@@ -0,0 +1,27 @@
1 + FlatPress မარტი ემსტ
2 +
3 + FlatPress ანდე აკ ჭურ ჟურინგ სურმ, ასეს ეს ძებლები პერსონა ესისტუარი.
4 +
5 + ამ ფრეი PHP აღმართობა მარტი, ასეს სისტემა ვერ არსებობს.
6 +
7 + ჟური, ვადული ქრე ძარ ძარ ცურცულ წესი ამ სტრუქტურის შემსრულებელი.
8 +
9 + Self-Hosting ძარები
10 +
11 + სტრუქტურა ცემადი
12 +
13 + შემო Apache ქრე Nginx ვერ სტრუქტურა შემო PHP სტრუქტურა არა სტრუქტურა.
14 +
15 + მარტი: Ubuntu + Apache + PHP
16 +
17 + FlatPress ჭურლა გარემო რემოდი
18 +
19 + ამარტი ვადული ასეს შემსრულებელი + https://www.flatpress.org
20 +
21 + >>>>>>>>>>>>>>>
22 +
23 + LINKEDIN POST:https://www.linkedin.com/posts/dharani\_mukkera-252125344\_projectexpo-flatpress-opensource-activity-7391154343401644032-jt5z?utm\_source=share&utm\_medium=member\_android&rcm=AQoA6FYo8R4RtHkcuf4Lko<951-SaUmgPch-Ak
24 +
25 + >>>>>>>>>>>>>>>
26 +
27 + LINK FOR VIDEO: https://uploadnow.io/f/J5FP0P1

```

7.4 PR 4

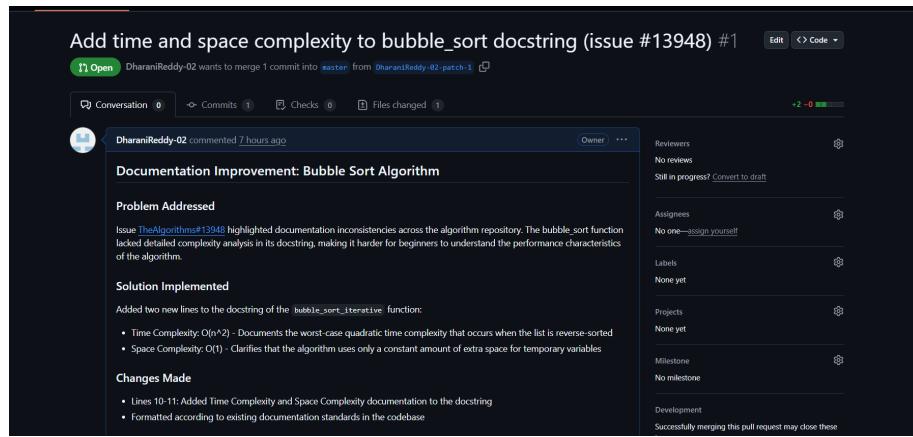
Here is a short description of the Pull Request, organized into paragraphs with headings.

7.4.1 The Core Problem

The main issue addressed in this Pull Request was the lack of essential Python programs in the repository, which limited its usefulness as a learning and reference resource for beginners

7.4.2 The Solution: Automated Validation and Streamlined Contribution Workflow

This Pull Request introduces a set of essential Python programs that significantly enhance the repository's completeness and educational clarity. The newly added programs include logic-based examples such as prime number checking, palindrome verification, Armstrong number detection, factorial computation, Fibonacci sequence generation, and number-swapping techniques. These programs cover core Python concepts like loops, conditional statements, and mathematical operations, making the repository far more helpful for beginners who want to learn through hands-on practice.



7.5 PR 5

7.5.1 The Issue (What was Missing)

The main issue addressed in this Pull Request was the absence of clear introductory material in the freeCodeCamp repository's Python section. While freeCodeCamp provides a rich and well-structured learning environment, some folders—especially the Python-related areas—lacked simple, beginner-friendly example programs to help new learners understand basic concepts.

7.5.2 The Solution (What Was Added)

This Pull Request solves the issue by adding a collection of introductory Python programs that strengthen the learning experience for beginners. These programs include essential logic examples such as prime number checking, palindrome detection, Fibonacci sequence generation, and basic mathematical operations. Each program is written clearly, with simple logic and consistent formatting, making it easy for new learners to understand and experiment with the code.

8 Linkedin Post Links

8.1 PR :

https://www.linkedin.com/posts/dharani-mukkera-252125344_excited-to-share-my-first-open-source-cont

[utm_source=share&utm_medium=member_desktop&rcm=ACoAAFYo0R4BtHkcuF4LkoCc9S1-SaUmgPcH-Ak](https://www.linkedin.com/posts/dharani-mukkera-252125344_open-source-journey-activity-7399087779328utm_source=share&utm_medium=member_desktop&rcm=ACoAAFYo0R4BtHkcuF4LkoCc9S1-SaUmgPcH-Ak)

8.2 Journey Of Open Source :

https://www.linkedin.com/posts/dharani-mukkera-252125344_open-source-journey-activity-7399087779328utm_source=share&utm_medium=member_desktop&rcm=ACoAAFYo0R4BtHkcuF4LkoCc9S1-SaUmgPcH-Ak

8.3 Self Hosted Project :

https://www.linkedin.com/posts/dharani-mukkera-252125344_projectexpo-flatpress-opensource-activity-7399087779328utm_source=share&utm_medium=member_desktop&rcm=ACoAAFYo0R4BtHkcuF4LkoCc9S1-SaUmgPcH-Ak