

Linux Privacy, Security, Self-Hosting, and Open-Source Contribution Report

Project Report Submission using LaTeX

Submitted To:

Dr. Arunekumar Bala
Open Source Engineering (HTE)
K L University



Submitted By:

Gummalla Jashnavi
B.Tech - CSE (Honors Through Experiential Learning)

Contents

1	Linux Distribution Used	3
1.1	Introduction	3
1.2	Features	3
1.3	System Requirements	3
1.4	Installation Process	3
1.5	Why I Chose Ubuntu	4
2	Encryption and GPG	5
2.1	Introduction	5
2.2	Types of Encryption	5
2.3	What is GPG?	5
2.4	Generate GPG Key	5
2.5	Encrypt a file	5
2.6	Decrypt a file	6
3	Sending Encrypted Email	7
3.1	Introduction	7
3.2	Steps	7
3.3	Encrypt Email	7
3.4	Decrypt Email	7
4	Self-Hosted Server -About, Installation, Telugu, Poster	8
4.1	About Self-Hosted Server	8
4.2	Benefits	8
4.3	Installation Steps	8
4.4	Use Cases	9
4.5	Localized Telugu Section	9
5	Poster	10
5.1	Poster Content	10
5.2	Poster Image	10

5.3	Points	11
6	Open Source Contributions	12
6.1	Contribution Summary	12
6.2	PR Status Table	12
6.3	Pull Request Details	13
7	Social Media Publications	17
7.1	LinkedIn Posts	17
7.1.1	Self Hosting Experience Post	17
7.1.2	PR Merge Contribution Post	17
7.1.3	Blog Article Post	17

Chapter 1

Linux Distribution Used

1.1 Introduction

The Linux distribution used for this project is **Ubuntu**, one of the most popular and user-friendly Linux operating systems. Ubuntu is based on Debian and provides a stable, secure, and regularly updated environment.

1.2 Features

- GNOME desktop environment
- APT package manager
- Large software repository
- Regular security updates
- Built-in encryption support

1.3 System Requirements

- 2 GB RAM (4 GB recommended)
- 20 GB storage
- Dual-core processor

1.4 Installation Process

1. Download Ubuntu ISO.

2. Create bootable USB.
3. Boot and install Ubuntu.
4. Update system: `sudo apt update & sudo apt upgrade`

1.5 Why I Chose Ubuntu

- Stable and secure
- Beginner friendly
- Best for GPG, encryption, privacy tools

Chapter 2

Encryption and GPG

2.1 Introduction

Encryption converts readable data (plaintext) into unreadable ciphertext to protect it from unauthorized access.

2.2 Types of Encryption

1. **Symmetric** – same key for encrypt/decrypt.
2. **Asymmetric** – public and private keys.

2.3 What is GPG?

GPG (GNU Privacy Guard) implements OpenPGP and provides:

- Encryption of files/messages
- Digital signatures
- Secure key management

2.4 Generate GPG Key

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

2.5 Encrypt a file

```
gpg --encrypt --recipient user@example.com file.txt
```

2.6 Decrypt a file

```
gpg --decrypt file.txt.gpg
```

Chapter 3

Sending Encrypted Email

3.1 Introduction

GPG email encryption ensures only the recipient can read your email.

3.2 Steps

1. Export your public key
2. Import recipient's public key
3. Encrypt message
4. Paste encrypted text in email

3.3 Encrypt Email

```
gpg --encrypt --armor --recipient user@example.com message.txt
```

3.4 Decrypt Email

```
gpg --decrypt message.txt.asc
```

Chapter 4

Self-Hosted Server -About, Installation, Telugu, Poster

4.1 About Self-Hosted Server

A self-hosted server is a server you run on your own system, rather than relying on cloud providers. It gives complete control over data, security, customization, and privacy.

4.2 Benefits

- Full control over data
- Zero cloud hosting fees
- High security and privacy
- Customizable environment
- Offline functionality
- Great for learning backend, DevOps
- Suitable for personal, academic, or small-team projects

4.3 Installation Steps

Follow these steps to set up your self-hosted server:

1. **Prepare your system:** Ensure you have a Linux environment (Ubuntu recommended) with administrative access.
2. **Install essential software:**

- Python 3.x – for backend scripts
- Node.js – for frontend or server-side JS
- Git – for cloning repositories
- MySQL/PostgreSQL – optional, for database support

3. Clone your project repository:

```
git clone <repo-url>
cd <project-folder>
```

4. Install project dependencies:

- Python dependencies: pip install -r requirements.txt
- Node.js dependencies: npm install

5. Configure environment variables (if required):

Create a .env file in your project folder with necessary credentials, ports, or API keys.

6. Start the server:

- Python-based server: python app.py
- Node.js server: npm start or node server.js

7. Access the server:

Open your browser and navigate to: <http://localhost:8000> (or the port configured in your project)

4.4 Use Cases

- Hosting personal applications
- Git or code servers
- Nextcloud, WingFit, or custom apps
- College labs or small LAN deployments

4.5 Localized Telugu Section

సల్వ్-పోస్ట్ సర్వీర్ అంటే మనం మన సిస్టమ్‌లోనే సర్వీర్ను రన్ చేయడం. దీనివల్ల డేటాపై 100% నియంత్రణ మన చేతుల్లో ఉంటుంది. క్లోడ్ ప్రావైడర్ల్పై ఆధారపడాల్సిన అవసరం ఉండదు. ఇది విద్యార్థులు, డెవలపర్లు మరియు చిన్న టీమ్లకు చాలా ఉపయోగకరంగా ఉంటుంది.

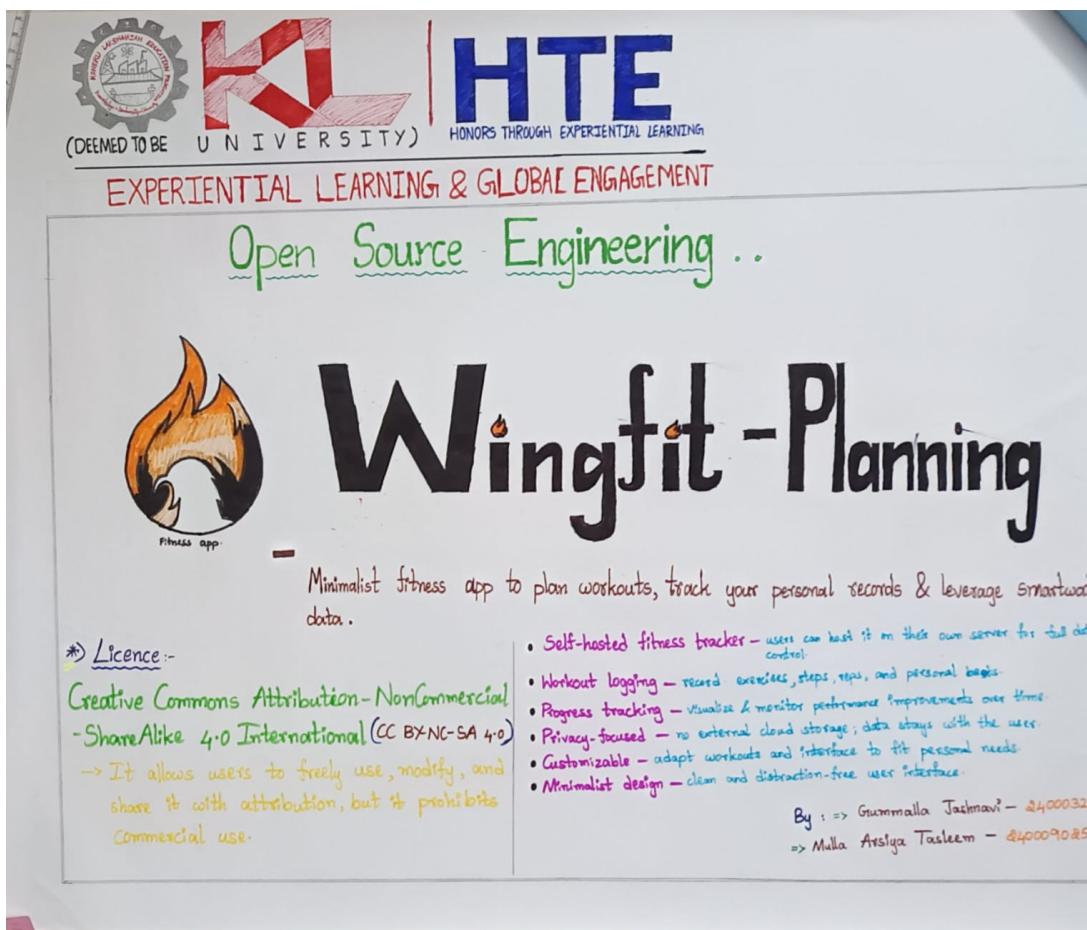
Chapter 5

Poster

5.1 Poster Content

WingFit – Self-Hosted Server
Secure • Fast • Local • Private

5.2 Poster Image



5.3 Points

- 100% Data Privacy
- Full Control Over Server
- Zero Cloud Hosting Fees
- Customizable Setup
- Offline Development Support
- Learn Backend and Deployment

Chapter 6

Open Source Contributions

During Hacktoberfest 2025, I actively participated in Open-Source projects by raising Pull Requests (PRs) to various repositories. This chapter documents all PRs, their purpose, status, and contribution details.

6.1 Contribution Summary

- Total Pull Requests Raised: 6
- Pull Requests Merged: 2
- Pull Requests Open / Pending: 4
- Number of repositories contributed to: 6

6.2 PR Status Table

PR No.	Repository	Status
1	DhanushNehru/Hacktoberfest2025	Merged
2	firstcontributions/first-contributions	Merged
3	KLGUG/Y24OpenSourceEngineering	Open
4	EbookFoundation/free-programming-books	Open
5	ServiceNowDevProgram/Hacktoberfest	Open
6	DHEERAJHARODE/Hacktoberfest2025-Open-source	Open

6.3 Pull Request Details

PR #1 – Added implement earliestTime method

Repository: DhanushNehru/Hacktoberfest2025

Status: Merged

Description: Implemented a new method to return earliest completion time.

Changes Done:

1. Added earliestTime() method
2. Solved edge case issues
3. Updated logic implementation

The screenshot shows a GitHub pull request page for PR #972. The title is "Added implement earliestTime method to find earliest completion". The status bar indicates it is "Merged" by "DhanushNehru" 2 weeks ago. The right sidebar shows the merge commit details: "Jashnavi25 commented 3 weeks ago" and "Join the Discussion!". The "PR Details" section includes a "Description of Changes" and a "Related Issues/Discussions" link. The "Contributor Info" section has a "Your Name" field. The right sidebar lists project details: Reviewers (No reviews), Assignees (No one assigned), Labels (None yet), Projects (None yet), Milestone (No milestone), Development (Successfully merging this pull request may close these issues. None yet), and Notifications (Customize). A "Merged" button is visible at the bottom right of the sidebar.

PR #1

PR #2 – Add name to Contributors list

Repository: firstcontributions/first-contributions

Status: Merged

Description: Added name entry to Contributors.md.

Changes Done:

1. Edited Contributors.md

Merged

PR #2

PR #3 – Wingfit Self-Hosting Documentation

Repository: KLGUG/Y24OpenSourceEngineering

Status: Open

Description: Added WingFit documentation and installation guide.

Changes Done:

1. Added guide document
2. Added setup instructions
3. Added LinkedIn post reference

Figure 6.1: Open PR #3

PR #4 – Update link for Python Full Course in Telugu

Repository: EbookFoundation/free-programming-books

Status: Open

Description: Updated YouTube course link to active version.

Changes Done:

1. Replaced outdated YouTube link

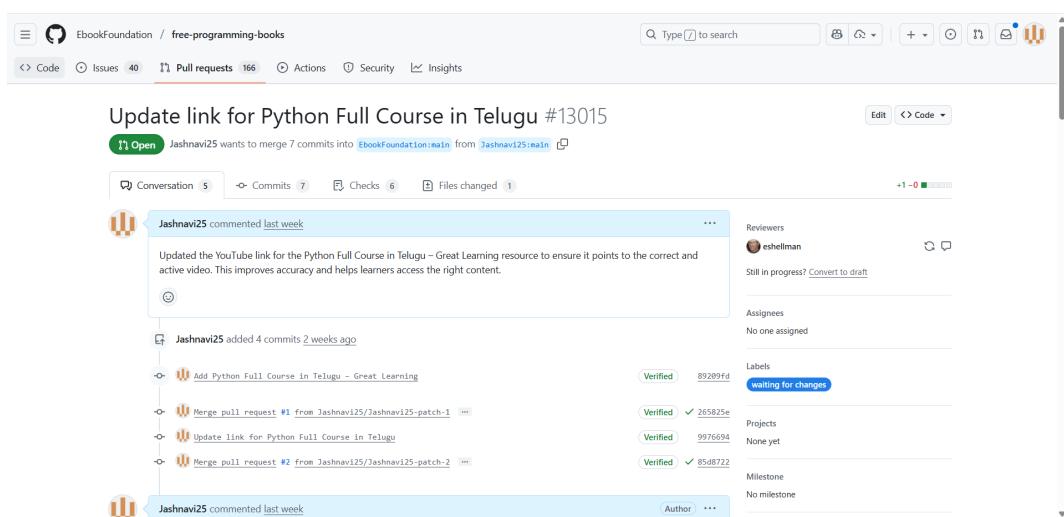


Figure 6.2: Open PR #4

PR #5 – Documentation grammar fixes

Repository: ServiceNowDevProgram/Hacktoberfest

Status: Open

Description: Improved grammar and readability in CONTRIBUTING.md.

Changes Done:

1. Grammar fixes
2. Readability improvements

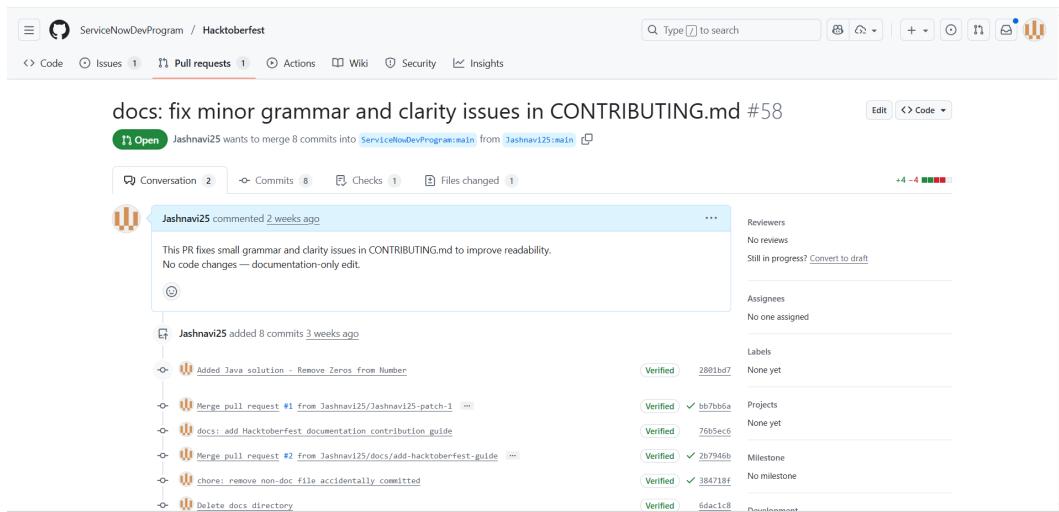


Figure 6.3: Open PR #5

PR #6 – Implement GCD functions

Repository: DHEERAJHARODE/Hacktoberfest2025-Open-source
Status: Open

Description: Implemented multiple mathematical GCD operations.

Changes Done:

1. Implemented GCD()
2. Added odd/even GCD logic

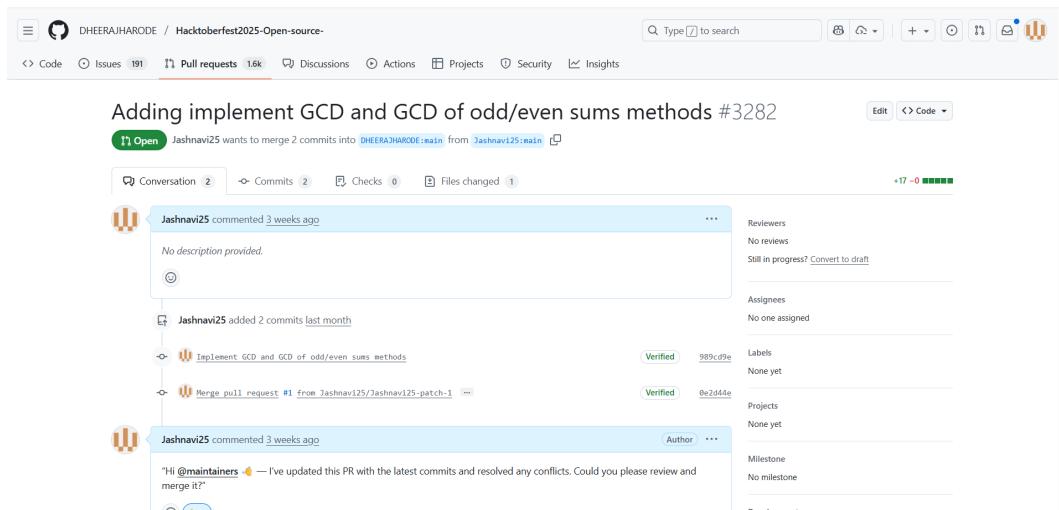


Figure 6.4: Open PR #6

Chapter 7

Social Media Publications

Below are the public posts I made related to the project, self-hosting, and open source contributions.

7.1 LinkedIn Posts

7.1.1 Self Hosting Experience Post

Link:

1. https://www.linkedin.com/posts/jashnavi-gummalla_opensource-kluniversity-foss-ugcPost-7382312489163628544-Rwk1?utm_source=share&utm_medium=member_desktop&rcm=ACoAAFgF3RQBrd4sKR1pEPoiRWZizcct29LrADA

7.1.2 PR Merge Contribution Post

Link:

2. https://www.linkedin.com/posts/jashnavi-gummalla_klu-hte-opensourceengineering-ugcPost-7390802921300467712-HqZg?utm_source=share&utm_medium=member_desktop&rcm=ACoAAFgF3RQBrd4sKR1pEPoiRWZizcct29LrADA

7.1.3 Blog Article Post

Link:

3. https://www.linkedin.com/posts/jashnavi-gummalla_its-about-my-experience-through-open-source-ugcPost-7398717143338622976-guAv?utm_source=share&utm_medium=member_desktop&rcm=ACoAAFgF3RQBrd4sKR1pEPoiRWZizcct29LrADA