Open Source Software

Student Details

Name:	DUVVU VENKATA RAMANA
Roll Number:	2400030646
Department:	Computer Science and Engineering
University:	KL University
Course:	Open Source
Semester:	ODD SEMESTER

Submitted to:

Dr. Sripath Roy
Department of Computer Science and Engineering
KL University

Contents

1 Linux Distribution **4**
1.1 Distribution Used: Ubuntu 22.04 LTS . . . 4
1.2 Why Ubuntu? . . . 4
1.3 Key Features of Ubuntu 22.04 LTS . . . 4
1.4 System Specifications . . . 4
1.5 Installation Process . . . 4

2 Encryption and GPG **5**
2.1 What is Encryption? . . . 5
2.2 Types of Encryption . . . 5
2.2.1 Symmetric Encryption . . . 5
2.2.2 Asymmetric Encryption . . . 5
2.3 GNU Privacy Guard (GPG) . . . 5
2.4 Installing GPG . . . 5
2.5 Generating GPG Keys . . . 6
2.6 Listing Keys . . . 6
2.7 Exporting Public Key . . . 6
2.8 Encrypting Files . . . 6
2.9 Decrypting Files . . . 6

3 Sending Encrypted Email **6**
3.1 Email Encryption Overview . . . 6
3.2 Tools Used . . . 6
3.3 Setting up Thunderbird with GPG . . . 7
3.3.1 Installation . . . 7
3.3.2 Configuring OpenPGP . . . 7
3.4 Sending Encrypted Email . . . 7
3.5 Receiving Encrypted Email . . . 7
3.6 Best Practices . . . 7

4 Privacy Tools from prism-break.org **8**
4.1 What is PRISM-Break? . . . 8
4.2 Tool 1: Signal - Encrypted Messaging . . . 8
4.3 Tool 2: Firefox - Web Browser . . . 8
4.4 Tool 3: ProtonMail - Encrypted Email . . . 9
4.5 Tool 4: Tor Browser - Anonymous Browsing . . . 9
4.6 Tool 5: VeraCrypt - Disk Encryption . . . 10

5 Open Source License **10**
5.1 License Used: MIT License . . . 10
5.2 What is the MIT License? . . . 10
5.3 MIT License Text . . . 10
5.4 Why Choose MIT License? . . . 11
5.5 Other Common Open Source Licenses . . . 11
5.5.1 GPL (GNU General Public License) . . . 11
5.5.2 Apache License 2.0 . . . 11

5.5.3 BSD License 11

6 Self-Hosted Server: QloApps 12
6.1 What is QloApps? 12
6.2 Team Members 12
6.3 Project Resources 12
6.4 Why Self-Host QloApps? 12
6.5 Installation Guide 13
6.5.1 System Requirements 13
6.5.2 Installation Steps 13
6.6 Key Features of QloApps 16
6.6.1 Hotel Management 16
6.6.2 Booking System 16
6.6.3 Customer Management 16
6.6.4 Payment Integration 16
6.7 Localization - Telugu Translation Documentation 16
6.7.1 Telugu Documentation Title 16
6.7.2 Key Sections Translated 17
6.7.3 Translation Highlights 17
6.7.4 Documentation Benefits 17
6.8 Benefits of Self-Hosting QloApps 17
6.9 Backup and Maintenance 18
6.9.1 Database Backup 18
6.9.2 Files Backup 18
6.9.3 Useful Resources 18

7 Open Source Contributions Overview 18
7.1 PR 1: Fix typo in offline_operations_file_does_not_exist 18
7.1.1 Issue 18
7.1.2 Solution 18
7.2 PR 2: Fix misleading Worker API documentation about daemon persistence 18
7.2.1 Issue 19
7.2.2 Solution 19
7.3 PR 3: Fix incorrect CLI output indentation in Problems API documentation 19
7.3.1 Issue 19
7.3.2 Solution 19
7.4 PR 4: Refine Telugu translation in README.te.md 19
7.4.1 Issue 19
7.4.2 Solution 19
7.5 PR 5: Update Contributors.md with new contributor 19
7.5.1 Issue 20
7.5.2 Solution 20
7.6 PR 6: Add FAQ reminder to bug report template 20
7.6.1 Issue 20
7.6.2 Solution 20
7.7 PR 7: Add missing source code editing shortcut 20
7.7.1 Issue 20
7.7.2 Solution 20

7.8 PR 8: Improve clarity and grammar in index troubleshooting frontmatter 20
7.8.1 Issue . . . 21
7.8.2 Solution . . . 21
7.9 PR 9: Fix #6147 – Wrap file paths in backticks . . . 21
7.9.1 Issue . . . 21
7.9.2 Solution . . . 21
7.10 PR 10: Fix Armstrong number implementation . . . 21
7.10.1 Issue . . . 21
7.10.2 Solution . . . 21
7.11 PR 11: Implement One-Time Pad cipher . . . 21
7.11.1 Issue . . . 22
7.11.2 Solution . . . 22
7.12 Key Learnings from Contributions . . . 22

8 LinkedIn Posts 22
8.1 Post 1: Contributor to Global Projects . . . 22
8.2 Post 2: FOSS Culture at KL University . . . 23
8.3 Post 3: Blog Link to My Open Source Journey . . . 23

9 Conclusion 23

1 Linux Distribution

1.1 Distribution Used: Ubuntu 22.04 LTS

For this project, I have used **Ubuntu 22.04 LTS** as my primary operating system.

1.2 Why Ubuntu?

Ubuntu is one of the most popular Linux distributions for several reasons:

- **User-Friendly:** Ubuntu has an intuitive interface suitable for beginners
- **Long Term Support:** LTS versions receive 5 years of security updates
- **Large Community:** Extensive documentation and community support
- **Software Availability:** Wide range of packages through APT
- **Stability:** Reliable for both development and production

1.3 Key Features of Ubuntu 22.04 LTS

1. **Desktop Environment:** GNOME 42
2. **Kernel Version:** Linux 5.15 LTS
3. **Package Manager:** APT (Advanced Package Tool)
4. **Default Applications:** Firefox, LibreOffice, GNOME utilities
5. **Snap Support:** Built-in support for snap packages

1.4 System Specifications

My system configuration:

- Operating System: Ubuntu 22.04 LTS
- Architecture: x86_64
- Desktop Environment: GNOME
- Shell: Bash 5.1

1.5 Installation Process

The installation involved:

1. Downloaded Ubuntu 22.04 LTS ISO from official website
2. Created bootable USB using Rufus/Etcher
3. Configured dual boot with existing OS
4. Installed essential development tools
5. Configured system for open source development

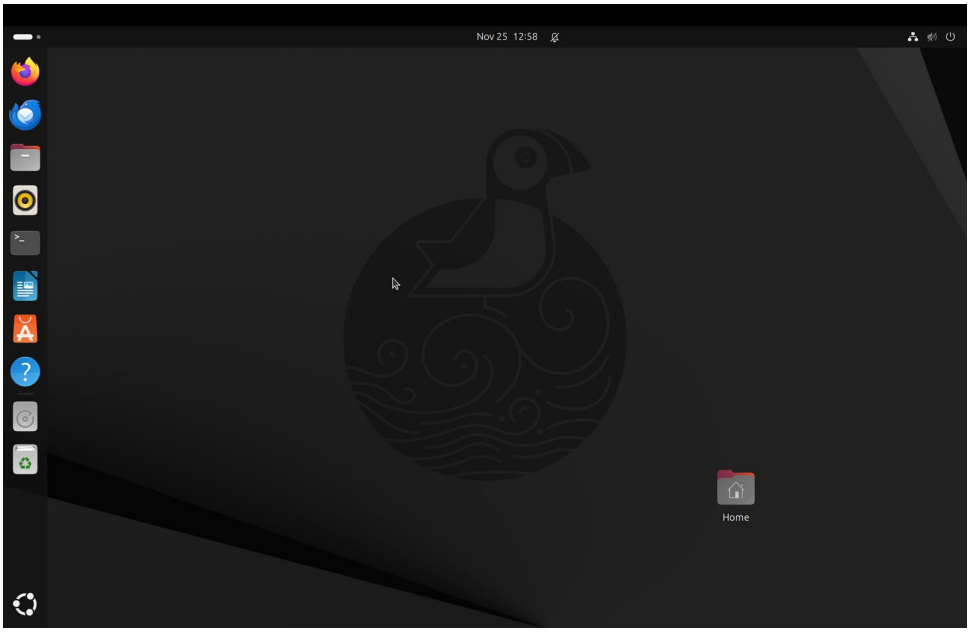

Figure 1: Ubuntu

2 Encryption and GPG

2.1 What is Encryption?

Encryption is the process of converting plaintext into ciphertext to protect data confidentiality. It ensures that only authorized parties can access the information.

2.2 Types of Encryption

2.2.1 Symmetric Encryption

Uses the same key for encryption and decryption. Examples: AES, DES.

2.2.2 Asymmetric Encryption

Uses a public-private key pair. Examples: RSA, ECC.

2.3 GNU Privacy Guard (GPG)

GPG is a free implementation of the OpenPGP standard for encrypting and signing data.

2.4 Installing GPG

```
sudo apt update
sudo apt install gnupg
gpg --version
```

2.5 Generating GPG Keys

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

Steps followed:

1. Selected RSA and RSA (default)
2. Key size: 4096 bits
3. Key validity: 1 year
4. Entered name and email
5. Created strong passphrase

2.6 Listing Keys

```
gpg --list-keys
gpg --list-secret-keys
```

2.7 Exporting Public Key

```
gpg --armor --export your-email@example.com > public-key.asc
```

2.8 Encrypting Files

```
gpg --encrypt --recipient your-email@example.com document.txt
```

2.9 Decrypting Files

```
gpg --decrypt document.txt.gpg > document.txt
```

3 Sending Encrypted Email

3.1 Email Encryption Overview

Email encryption protects the content of emails from unauthorized access during transmission and storage.

3.2 Tools Used

- **Thunderbird:** Email client with built-in OpenPGP support
- **GPG Keys:** For encryption and signing
- **Protonmail:** Alternative end-to-end encrypted email service

3.3 Setting up Thunderbird with GPG

3.3.1 Installation

```
sudo apt install thunderbird
```

3.3.2 Configuring OpenPGP

Steps followed:

1. Open Thunderbird
2. Go to Account Settings
3. Select End-to-End Encryption
4. Add existing GPG key or generate new one
5. Import recipient's public key

3.4 Sending Encrypted Email

Process:

1. Compose new email
2. Click on Security button
3. Select "Require Encryption"
4. Optionally add digital signature
5. Send email

3.5 Receiving Encrypted Email

When receiving:

1. Email appears encrypted
2. Thunderbird automatically detects encryption
3. Enter GPG passphrase
4. Email content is decrypted and displayed

3.6 Best Practices

- Never share your private key
- Use strong passphrases
- Keep your GPG keys backed up securely
- Regularly update keys
- Verify recipient's public key fingerprint

4 Privacy Tools from prism-break.org

4.1 What is PRISM-Break?

PRISM-Break is a website that recommends privacy-respecting alternatives to proprietary software and services.

4.2 Tool 1: Signal - Encrypted Messaging

Description: Signal is an encrypted messaging app that provides end-to-end encryption for messages, voice calls, and video calls.

Key Features:

- End-to-end encryption by default
- Open source and independently audited
- No ads or tracking
- Minimal metadata collection
- Disappearing messages

Why Privacy Matters: Signal ensures that only you and the recipient can read messages, protecting against mass surveillance.

4.3 Tool 2: Firefox - Web Browser

Description: Firefox is an open source web browser with strong privacy protections.

Privacy Features:

- Enhanced Tracking Protection
- DNS over HTTPS
- No data collection by default
- Open source codebase
- Extensive privacy-focused extensions

Configuration Tips:

- Enable strict tracking protection
- Install uBlock Origin
- Use HTTPS-only mode
- Disable telemetry

4.4 Tool 3: ProtonMail - Encrypted Email

Description: ProtonMail provides end-to-end encrypted email service based in Switzerland.

Key Features:

- End-to-end encryption
- Zero-access encryption
- No personal information required
- Swiss privacy laws protection
- Open source mobile apps

Use Cases:

- Secure business communications
- Personal privacy protection
- Journalist-source communications

4.5 Tool 4: Tor Browser - Anonymous Browsing

Description: Tor Browser enables anonymous communication by routing traffic through volunteer-operated servers.

How It Works:

- Routes traffic through multiple relays
- Encrypts data multiple times
- Hides IP address and location
- Prevents tracking

Best Use Cases:

- Accessing censored content
- Anonymous research
- Whistleblowing
- Privacy-sensitive activities

4.6 Tool 5: VeraCrypt - Disk Encryption

Description: VeraCrypt is a free open source disk encryption software.

Features:

- Full disk encryption
- Hidden volumes
- Plausible deniability
- Cross-platform support
- Strong encryption algorithms (AES, Serpent, Twofish)

Use Cases:

- Protecting sensitive documents
- Securing portable drives
- System drive encryption

5 Open Source License

5.1 License Used: MIT License

For my open source contributions and projects, I primarily work with the **MIT License**.

5.2 What is the MIT License?

The MIT License is a permissive free software license that allows users to:

- Use the software commercially
- Modify the software
- Distribute the software
- Use the software privately
- Sublicense the software

5.3 MIT License Text

```
MIT License

Copyright (c) 2025 Duvvu Venkata Ramana

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
```

```
copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND...
```

5.4 Why Choose MIT License?

1. **Simple and Easy:** Short and easy to understand
2. **Permissive:** Minimal restrictions on reuse
3. **Business-Friendly:** Can be used in proprietary software
4. **Popular:** Widely used and recognized
5. **Compatible:** Works well with other licenses

5.5 Other Common Open Source Licenses

5.5.1 GPL (GNU General Public License)

- Copyleft license
- Requires derivative works to be open source
- Used by Linux kernel

5.5.2 Apache License 2.0

- Permissive like MIT
- Includes patent grant
- Used by Apache projects

5.5.3 BSD License

- Very permissive
- Similar to MIT
- Used by FreeBSD

6 Self-Hosted Server: QloApps

6.1 What is QloApps?

QloApps is a free, open-source hotel management and online booking system built on PrestaShop that allows you to manage:

- Hotel rooms and properties
- Online and offline reservations
- Pricing and availability
- Customer management
- Website booking engine
- Payment processing

6.2 Team Members

This project was completed in collaboration with:

- **Venkata Ramana** - Lead Developer
- **Sunil** - Developer and Tester

6.3 Project Resources

- **Demo Video:** `https://drive.google.com/file/d/1hf5ehgALdg1_b7oY3Xf3rbdOR6PAqmsb/view?usp=drivesdk`
- **LinkedIn Post:** `https://www.linkedin.com/posts/venkata-ramana-80a137369_opensource-opensource-kluniversity-activity-7382319162603282433-hZ7D`

6.4 Why Self-Host QloApps?

1. **Full Data Control:** All customer and booking data stays on your server
2. **Customization:** Modify templates, modules, pricing logic
3. **Cost Effective:** Core system is free and open source
4. **Independence:** No reliance on third-party hotel software subscriptions
5. **Learning Experience:** Install, configure, and maintain a real Property Management System

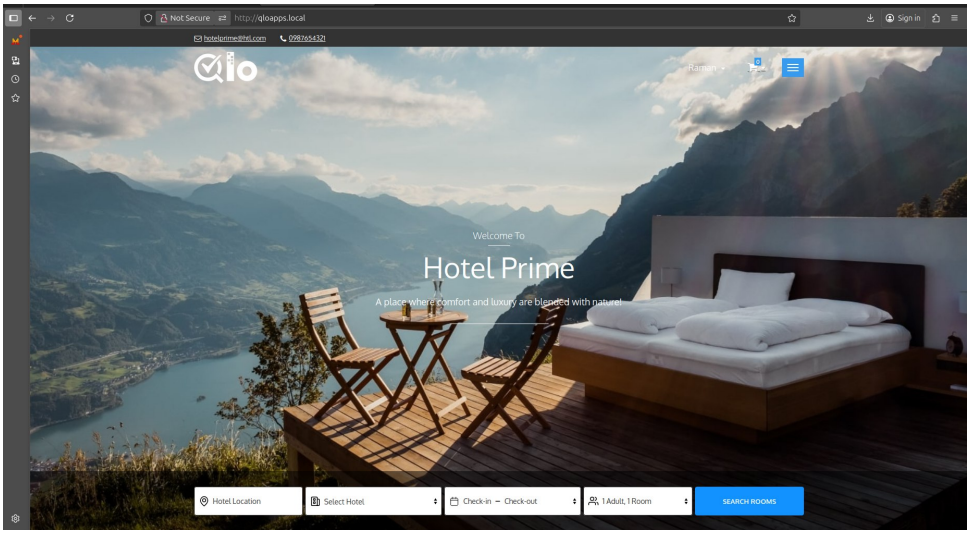

Figure 2: QloApps Dashboard

Component	Recommended Version
Operating System	Ubuntu 20.04 / 22.04 (64-bit)
Web Server	Apache 2.4 or higher
PHP	7.4 – 8.1
Database	MariaDB / MySQL 5.7 or higher
RAM	Minimum 2 GB
Storage	Minimum 2 GB free space

Table 1: QloApps System Requirements

6.5 Installation Guide

6.5.1 System Requirements

6.5.2 Installation Steps

Step 1: Update System and Install Required Packages

```
sudo apt update && sudo apt upgrade -y
sudo apt install apache2 -y
sudo apt install mariadb-server mariadb-client -y
sudo apt install php php-cli libapache2-mod-php php-mysql \
  php-curl php-gd php-mbstring php-xml php-zip php-intl \
  php-bcmath unzip -y
sudo a2enmod rewrite
sudo systemctl restart apache2
```

Step 2: Secure MariaDB

```
sudo mysql_secure_installation
```

Recommended answers:

Step 3: Create QloApps Database

```
sudo mariadb
```

Execute these SQL commands:

Question	Answer
Switch to unix_socket authentication	Y
Remove anonymous users	Y
Disallow root login remotely	Y
Remove test database and access	Y
Reload privilege tables now	Y

```
CREATE DATABASE qloappsdb;
CREATE USER 'qloappsuser'@'localhost' IDENTIFIED BY '
   StrongPassword123!';
GRANT ALL PRIVILEGES ON qloappsdb.* TO 'qloappsuser'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

Step 4: Download QloApps

```
cd /tmp
wget https://github.com/Qloapps/QloApps/archive/refs/tags/v1.7.0.
   zip -O qloapps.zip
unzip qloapps.zip
sudo mv QloApps-1.7.0 /var/www/html/qloapps
```

Step 5: Set Folder Permissions

```
sudo chown -R www-data:www-data /var/www/html/qloapps
sudo chmod -R 755 /var/www/html/qloapps
sudo chmod -R 777 /var/www/html/qloapps/upload
sudo chmod -R 777 /var/www/html/qloapps/download
sudo chmod -R 777 /var/www/html/qloapps/cache
sudo chmod -R 777 /var/www/html/qloapps/logs
```

Step 6: Configure Apache Virtual Host

```
sudo nano /etc/apache2/sites-available/qloapps.conf
```

Add this configuration:

```
<VirtualHost *:80>
    ServerAdmin admin@localhost
    DocumentRoot /var/www/html/qloapps
    ServerName qloapps.local

    <Directory /var/www/html/qloapps/>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/qloapps_error.log
    CustomLog ${APACHE_LOG_DIR}/qloapps_access.log combined
</VirtualHost>
```

Enable the site:

```
sudo a2ensite qloapps.conf
sudo systemctl restart apache2
```

Add to /etc/hosts:

```
127.0.0.1 qloapps.local
```

Step 7: Browser Installation
Open in browser:

```
http://localhost/qloapps
```

or

```
http://qloapps.local
```

Follow the installation wizard:

1. Select language
2. Accept license
3. Enter database details:
   - Database name: qloappsdb
   - User: qloappsuser
   - Password: StrongPassword123!
4. Complete installation

Step 8: Post-Installation
Remove install folder:

```
sudo rm -rf /var/www/html/qloapps/install
```

Access admin panel:

```
http://localhost/qloapps/admin
```

Optional PHP settings:

```
sudo nano /etc/php/8.1/apache2/php.ini
```

Modify:

```
upload_max_filesize = 16M
post_max_size = 16M
max_execution_time = 300
```

Restart Apache:

```
sudo systemctl restart apache2
```

6.6 Key Features of QloApps

6.6.1 Hotel Management

- Multiple property management
- Room type configuration
- Pricing and availability calendar
- Seasonal pricing rules
- Room inventory tracking

6.6.2 Booking System

- Online booking engine
- Offline/manual bookings
- Booking modifications
- Cancellation management
- Booking status tracking

6.6.3 Customer Management

- Guest profiles
- Booking history
- Customer communications
- Loyalty programs

6.6.4 Payment Integration

- Multiple payment gateways
- Online payment processing
- Invoice generation
- Payment tracking

6.7 Localization - Telugu Translation Documentation

As part of making QloApps accessible to Telugu-speaking users, I have created comprehensive documentation in Telugu language.

6.7.1 Telugu Documentation Title

QloApps
(QloApps Self-Hosting Documentation)

6.7.2 Key Sections Translated

The Telugu documentation includes:

1. **(Introduction):** Overview of QloApps and its capabilities
2. **(System Requirements):** Hardware and software requirements
3. **(Installation):** Step-by-step installation guide
4. **(Database Setup):** Database configuration instructions
5. **(Maintenance):** Backup and maintenance procedures

6.7.3 Translation Highlights

Key technical terms translated to Telugu:

English	Telugu (English Format)
Installation	Instalēṣan
Database	Dēṭābēs
Server	Sarvar
Requirements	Avsarālu
Configuration	Kānphigārēṣan
Backup	Byākap
Maintenance	Meṇṭenens

Table 2: English to Telugu Technical Terms with Transliteration

6.7.4 Documentation Benefits

- Makes QloApps accessible to Telugu-speaking hotel owners
- Removes language barriers in technology adoption
- Promotes open source software in regional languages
- Helps local businesses implement self-hosted solutions
- Demonstrates importance of localization in software

6.8 Benefits of Self-Hosting QloApps

1. **Complete Privacy:** All guest data remains on your server
2. **No Commission Fees:** Unlike booking platforms, no per-booking fees
3. **Full Customization:** Modify any aspect of the system
4. **Data Ownership:** Complete control over all business data
5. **Educational Value:** Learn hotel management systems and web hosting
6. **Cost Savings:** No monthly subscription fees
7. **Scalability:** Expand as your business grows

6.9 Backup and Maintenance

6.9.1 Database Backup

```
mysqldump -u root -p qloappsdb > qloapps_backup.sql
```

6.9.2 Files Backup

```
tar -czvf qloapps_files_backup.tar.gz /var/www/html/qloapps
```

6.9.3 Useful Resources

- Official Documentation: `https://devdocs.qloapps.com`
- Demo Website: `https://demo.qloapps.com`
- Forum / Support: `https://forums.qloapps.com`

This section details all pull requests I have contributed to various open source projects, including screenshots and detailed descriptions of each contribution.

7 Open Source Contributions Overview

7.1 PR 1: Fix typo in offline_operations_file_does_not_exist

Repository: nextcloud/android
PR Number: #16008
Status: Merged

7.1.1 Issue

A typo in the strings.xml file caused incorrect text to appear in the offline operations message.

7.1.2 Solution

- Corrected the typo in strings.xml.
- Improved language accuracy in the UI.

7.2 PR 2: Fix misleading Worker API documentation about daemon persistence

Repository: gradle/gradle
PR Number: #35781
Status: Merged

7.2.1 Issue

The Worker API documentation incorrectly described daemon persistence behavior.

7.2.2 Solution

- Updated misleading documentation lines.
- Clarified daemon lifecycle and Worker API usage.

7.3 PR 3: Fix incorrect CLI output indentation in Problems API documentation

Repository: gradle/gradle
PR Number: #35780
Status: Merged

7.3.1 Issue

CLI examples in the Problems API documentation had incorrect indentation.

7.3.2 Solution

- Corrected indentation across CLI examples.
- Improved readability and formatting consistency.

7.4 PR 4: Refine Telugu translation in README.te.md

Repository: firstcontributions/first-contributions
PR Number: #106725
Status: Merged

7.4.1 Issue

Telugu translation contained unclear phrases and inconsistent terminology.

7.4.2 Solution

- Refined phrasing for better readability.
- Standardized terminology.

7.5 PR 5: Update Contributors.md with new contributor

Repository: firstcontributions/first-contributions
PR Number: #106696
Status: Merged

7.5.1 Issue

The contributor list required updates to include new members.

7.5.2 Solution

- Added the new contributor entry to Contributors.md.

7.6 PR 6: Add FAQ reminder to bug report template

Repository: nlohmann/json
PR Number: #5015
Status: Open

7.6.1 Issue

Users submitted duplicate bug reports without checking the FAQ first.

7.6.2 Solution

- Added an FAQ reminder to the bug report template.
- Reduced duplicate issue submissions.

7.7 PR 7: Add missing source code editing shortcut

Repository: github/docs
PR Number: #41515
Status: Open

7.7.1 Issue

A keyboard shortcut for source code editing was missing from documentation.

7.7.2 Solution

- Added the missing shortcut.
- Improved completeness of the shortcuts guide.

7.8 PR 8: Improve clarity and grammar in index troubleshooting frontmatter

Repository: github/docs
PR Number: #41514
Status: Open

7.8.1 Issue

The troubleshooting index had unclear wording and grammar mistakes.

7.8.2 Solution

- Improved grammar and clarity.
- Enhanced readability for users.

7.9 PR 9: Fix #6147 – Wrap file paths in backticks

Repository: streamlit/streamlit
PR Number: #13095
Status: Open

7.9.1 Issue

File paths were not wrapped in backticks, causing Markdown to escape characters incorrectly.

7.9.2 Solution

- Wrapped paths in backticks.
- Ensured correct markdown rendering.

7.10 PR 10: Fix Armstrong number implementation

Repository: TheAlgorithms/Java
PR Number: #7097
Status: Open

7.10.1 Issue

The Armstrong number implementation produced incorrect results.

7.10.2 Solution

- Corrected logic for Armstrong number calculation.
- Ensured accuracy through added tests.

7.11 PR 11: Implement One-Time Pad cipher

Repository: TheAlgorithms/Java
PR Number: #7096
Status: Open

7.11.1 Issue

The One-Time Pad cipher algorithm was missing from the repository.

7.11.2 Solution

- Implemented encryption and decryption methods.
- Added examples and documentation.

Metric	Count
Total Pull Requests	11
Merged PRs	5
Open PRs	6
Repositories Contributed	7
Documentation Improvements	7
Code Contributions	2
Localization Work	2

Table 3: Open Source Contribution Statistics

7.12 Key Learnings from Contributions

1. **Git Workflow:** Mastered branching, committing, and pull request processes
2. **Code Review:** Learned to receive and implement feedback constructively
3. **Documentation:** Understood the importance of clear, comprehensive documentation
4. **Collaboration:** Experienced working with global developer communities
5. **Testing:** Learned to write comprehensive test suites
6. **Localization:** Appreciated the value of making software accessible in regional languages

8 LinkedIn Posts

8.1 Post 1: Contributor to Global Projects

Link: `https://www.linkedin.com/posts/venkata-ramana-80a137369_opensource-developerjou`
`utm_source=share&utm_medium=member_desktop&rcm=ACoAAFtSgsoBdSJBSij_BaJkbe4oWVEYO3AA1F`

Summary: Shared experiences contributing to some repository, focusing on implementing comprehensive test suites and best practices in software testing.

Key Points:

- Importance of thorough testing in open source
- Test-driven development approach

- Learning from code reviews
- Contributing to educational repositories

8.2 Post 2: FOSS Culture at KL University

Link: `https://www.linkedin.com/posts/venkata-ramana-80a137369_opensource-opensource-k`
`utm_source=share&utm_medium=member_desktop&rcm=ACoAAFtSgsoBdSJBSij_BaJkbe4oWVEYO3AA1F`

Summary: Discussed the Free and Open Source Software culture at KL University and encouraged peers to participate in open source.

Highlights:

- FOSS courses and initiatives at university
- Personal journey in open source
- Community-driven development model
- Real-world learning experience

8.3 Post 3: Blog Link to My Open Source Journey

Link: `https://www.linkedin.com/pulse/beginners-journey-open-source-venkata-ramana-5sc`

Summary: Shared a comprehensive blog post documenting my open source journey, including challenges faced and lessons learned.

9 Conclusion

This report documents my comprehensive journey in open source software development, including:

- Setting up development environment with Ubuntu Linux
- Understanding encryption and privacy tools
- Self-hosting QloApps for hotel management
- Making meaningful contributions to open source projects
- Sharing knowledge through professional networking

Through these experiences, I have gained:

1. Technical skills in multiple programming languages and systems
2. Understanding of collaborative development workflows
3. Experience with version control and code review processes
4. Appreciation for open source philosophy and community
5. Professional networking and communication abilities

6. Practical knowledge of server administration and web hosting

The experience of self-hosting QloApps has been particularly valuable, providing hands-on experience with:

- LAMP stack configuration
- Database management
- Web server administration
- Security best practices
- Real-world application deployment

Open source contribution has been an invaluable learning experience, providing real-world software development exposure and connecting me with a global community of developers committed to building better, more accessible software for everyone.