❖ <u>None disclosure</u>

A Non-Disclosure Agreement (NDA) is a legal contract between two or more parties that outlines the confidentiality of shared information. When a software developer joins a software firm, signing an NDA ensures that sensitive information about the company's products, strategies, and other confidential data remains protected.

➤ **Definition of Confidential Information**

This clause specifies what is considered confidential. It usually includes:

- Technical information like software code, algorithms, and designs.
- Business information such as strategies, financial data, and customer lists.
- Any other information the company deems confidential.

Simple Terms: Anything you learn about the company that isn't public knowledge should be kept secret.

➤ <u>Obligations of the Receiving Party</u>

This outlines the responsibilities of the person receiving the confidential information (in this case, the developer). It typically requires:

- Keeping the information confidential.
- Not using the information for any purpose other than what's agreed upon.
- Taking reasonable steps to protect the information from unauthorized access.
- ❖ **Simple Terms**: Don't tell anyone about the company's secrets and use them only for your job.

➢ Exclusions from Confidential Information

Not all information is protected under the NDA. This clause lists what is not considered confidential, such as:

- Information that is already public.
- Information the developer knew before joining the company.
- Information that becomes public through no fault of the developer.
- Information independently developed by the developer without using the company's confidential information.

Simple Terms: If the info is already out there, you knew it before, or you figured it out on your own, it's not covered by the NDA.

➢ Term and Termination

This clause specifies how long the NDA is in effect. It can be for a set number of years or indefinite until the confidential information no longer qualifies as confidential.

Simple Terms: You have to keep the company's secrets for a certain amount of time, even after you leave the job.

➢ Return of Materials

When the NDA ends, the receiving party must return or destroy any physical or digital materials containing confidential information.

Simple Terms: When you leave the company or the NDA ends, give back or destroy any confidential documents you have.

➢ No License

This clause clarifies that signing the NDA does not grant the developer any rights to the company's intellectual property or other confidential information, except for the limited purpose of their job.

Simple Terms: Just because you know the company's secrets doesn't mean you own them.

➢ Remedies

This explains the actions the company can take if the NDA is breached, such as seeking injunctions (court orders to stop the breach) or financial compensation for damages.

Simple Terms: If you break the NDA, the company can take you to court and seek compensation.

➢ Governing Law

This specifies which state or country's laws will apply to the NDA.

Simple Terms: If there's a legal dispute, it will be handled under specific state or country's laws.

➢ Entire Agreement

This clause states that the NDA is the complete agreement between the parties regarding confidentiality and supersedes all prior agreements.

Simple Terms: This document is the final word on confidentiality between you and the company.

➢ Sample Agreement (Hypothetical Firm: Tech Solutions Inc.)

Confidential Information

"Confidential Information" includes all non-public information disclosed by Tech Solutions Inc., whether written, oral, or otherwise, including but not limited to software code, technical designs, and business strategies.

Obligations

The developer agrees to (a) keep the Confidential Information confidential; (b) not disclose it to any third party without the written consent of Tech Solutions Inc.; and (c) use the Confidential Information solely for the purpose of performing their duties at Tech Solutions Inc.

Exclusions

Confidential Information does not include information that (a) is or becomes publicly known through no breach of this Agreement; (b) was known by the developer before disclosure by Tech Solutions Inc.; (c) is independently developed by the developer without use of or reference to Confidential Information.

Term and Termination

This Agreement will remain in effect for 5 years from the date of termination of the developer's employment with Tech Solutions Inc.

Return of Materials

Upon termination of this Agreement or upon request by Tech Solutions Inc., the developer will return or destroy all materials containing Confidential Information.

No License

Nothing in this Agreement grants the developer any rights to or under any of Tech Solutions Inc.'s patents, copyrights, or other intellectual property.

Remedies

The developer acknowledges that any breach of this Agreement may cause irreparable harm to Tech Solutions Inc., entitling Tech Solutions Inc. to seek injunctive relief and damages.

Governing Law

This Agreement will be governed by the laws of the State of California.

Entire Agreement

This Agreement constitutes the entire agreement between the parties with respect to the subject matter hereof and supersedes all prior or contemporaneous agreements.

Simple Terms for Sample Agreement:

- Keep the company's secrets and use them only for your job.
- If the info is already public or you knew it before, it's not a secret.
- Keep the secrets for 5 years after you leave the job.
- Return or destroy any secret documents when asked.
- Knowing the company's secrets doesn't mean you own them.
- If you break this agreement, the company can take legal action.
- If there's a dispute, California law applies.
- This document is the final word on keeping the company's secrets.

➤ SAMPLE AGGREMENT OF NON- DISCLOSURE

NON-DISCLOSURE AGREEMENT

PARTIES

- This Non-Disclosure Agreement (hereinafter referred to as the **"Agreement"**) is entered into on _____ (the **"Effective Date"**), by and between _____, with an address of _____, (hereinafter referred to as the **"Disclosing Party"**) and _____, with an address of _____, (hereinafter referred to as the **"Receiving Party"**) (collectively referred to as the **"Parties"**).

CONFIDENTIAL INFORMATION

- The Receiving Party agrees not to disclose, copy, clone, or modify any confidential information related to the Disclosing Party and agrees not to use any such information without obtaining consent.

- "Confidential information" refers to any data and/or information that is related to the Disclosing Party, in any form, including, but not limited to, oral or written. Such confidential information includes, but is not limited to, any information related to the business or industry of the Disclosing Party, such as discoveries, processes, techniques, programs, knowledge bases, customer lists, potential customers, business partners, affiliated partners, leads, know-how, or any other services related to the Disclosing Party.

RETURN OF CONFIDENTIAL INFORMATION

- The Receiving Party agrees to return all the confidential information to the Disclosing Party upon the termination of this Agreement.

OWNERSHIP

- This Agreement is not transferable and may only be transferred by written consent provided by both Parties.

GOVERNING LAW

- This Agreement shall be governed by and construed in accordance with the laws of _____.

SIGNATURE AND DATE

- The Parties hereby agree to the terms and conditions set forth in this Agreement and such is demonstrated by their signatures below:

➢ <u>What is Open Source Software?</u>

Open source software (OSS) is software with source code that anyone can inspect, modify, and enhance. It promotes transparency, collaboration, and community-driven development, allowing users and developers to freely use, study, modify, and distribute the software.

➢ **Terms and Conditions Associated with Open Source Software**

Open source software is governed by various licenses, each with specific terms and conditions. Here are some common ones:

➢ **GNU General Public License (GPL)**

Rights: Users can use, modify, and distribute the software.

Conditions: Modified versions must also be released under the GPL, ensuring all derivative works remain open source.

➢ **MIT License**

Rights: Users can use, copy, modify, merge, publish, distribute, sublicense, and sell the software.

Conditions: The license must be included in all copies or substantial portions of the software.

➢ **Apache License 2.0**

Rights: Users can use, modify, and distribute the software, with additional protections regarding patents.

Conditions: Modifications must be documented, and the license must be included in all copies. Contributors grant an explicit patent license.

➢ **BSD License**

Rights: Users can use, modify, and distribute the software.

Conditions: Redistribution of the source code must retain the original copyright notice and disclaimers. Modified versions must also include a note indicating that changes have been made.

➢ **Can You Contribute to Open Source Software?**

Yes, anyone can contribute to open source software.

1. **Find a Project**

- Use platforms like GitHub, GitLab, or Bitbucket to find open source projects that interest you.
- Look for projects with clear contribution guidelines and a welcoming community.

2. **Understand the Project**

- Read the project's README file, documentation, and contribution guidelines.
- Familiarize yourself with the codebase by exploring the source code and any existing issues or feature requests.

3. **Set Up Your Development Environment**

- Fork the repository to create your own copy of the project.
- Clone your fork to your local machine and set up the development environment according to the project's instructions.

4. **Make Contributions**

- **Bug Fixes:** Identify and fix bugs. Check the issue tracker for reported bugs that need attention.
- **New Features:** Implement new features that are in demand or suggested by the community.
- **Documentation:** Improve or update the documentation to make it easier for others to use and contribute to the project.
- **Code Reviews:** Review and provide feedback on pull requests submitted by other contributors.

5. **Submit Your Changes**

- Create a new branch for your changes.
- Make your changes in the new branch and commit them with clear, descriptive messages.
- Push your branch to your forked repository.

- Submit a pull request to the original repository, explaining the changes you've made and why they should be merged.

6. **Engage with the Community**

- Respond to feedback and questions from project maintainers and other contributors.
- Participate in discussions and contribute to the project's roadmap and future development.

Example Contribution Workflow

Fork the Repository: Click the "Fork" button on the project's GitHub page to create your own copy of the repository.
Clone the Forked Repository:
git clone https://github.com/your-username/project-name.git

Create a New Branch
cd project-name git checkout -b my-feature-branch

Make Changes: Edit the code, fix bugs, add features, or update documentation.
git add .git commit -m "Description of changes"

Push Changes to Your Fork:
git push origin my-feature-branch

Create a Pull Request:

Go to the original repository and submit a pull request from your forked repository.

By following these steps and adhering to the project's contribution guidelines, you

can effectively contribute to open source software and become part of the

community that maintains and improves it.

We can contribute in below projects like

- Linux Kernel   Blender
- Mozilla Firefox   WordPress
- LibreOffice   Audacity
- Apache HTTP Server   Kubernetes
- GIMP (GNU Image Manipulation Program)
- VLC Media Player