

CONCEITOS BÁSICOS DE GNU LINUX

Formas de contribuir em 11 formas iniciais e engrandecer em sua área de programação e isso ajuda bastante!

Comunidades de software livre e código aberto sempre querem contribuidores. Qualquer pessoa pode contribuir para diversos projetos desta natureza, independentemente do seu nível de conhecimento. É comum existirem contribuições anônimas ou registradas nas bases de dados dos projetos. Existem diferentes maneiras de contribuição nesses projetos, são elas:

- **Empacotamento de software.** Um pacote é uma coleção de arquivos que permite aplicativos ou bibliotecas serem distribuídos através do sistema de gerenciamento de pacotes. O objetivo de criar pacotes é permitir a automação da instalação, atualização, configuração e remoção de programas para os projetos de forma consistente e precisa. Um pacote normalmente consiste em um componente fonte e um ou mais componentes binários.
- **Instale e use o sistema operacional ou ferramenta.** Ao instalar e usar, você terá testado o instalador, aumentará a base de usuários e divulgação, além de ficar familiarizado com os sistemas.
- **Reportar falhas.** Ao encontrar uma falha nos sistemas, envie um *bug report* (relatório de erros) para que os Desenvolvedores fiquem sabendo e possam corrigi-lo. É possível acompanhar o ciclo das falhas reportadas.
- **Enviar correções.** Se você sabe como resolver um *bug*, você pode enviar uma requisição de conserto. Os desenvolvedores irão analisar e se tudo estiver correto, eles aplicarão a sua solução.
- **Documentação.** Você pode produzir documentação escrevendo manuais, tutoriais, *HOWTOs*, *FAQs*. Diferentes níveis de documentos como iniciante, intermediário ou avançado. Publicação escrita ou em vídeo.
- **Suporte a outros usuários.** Ajude a tirar dúvidas de outros usuários. Existem muitos locais que você pode participar como listas de e-mails, canais no IRC, fóruns e grupos do Facebook, Telegram.
- **Divulgação.** Ajudar a divulgar os sistemas em escolas, universidades, congressos, encontros, redes sociais, etc. Comprar produtos das lojas oficiais que ajudam na manutenção dos projetos.

●**Publicidade.** Os projetos possuem um time de publicidade que elabora textos de notícias e mantém alguns sites e perfis de redes sociais. Você pode contribuir com o time de publicidade ajudando a escrever as notícias ou a revisar os textos.

●**Organização de eventos.** Esses eventos podem ser desde um encontro em um bar ou restaurante até um evento com palestras e oficinas relacionadas aos projetos. O importante é promover encontros da comunidade local para celebrar o projeto.

●**Produção de material gráfico.** Você pode produzir materiais gráficos e disponibilizá-los para que outras pessoas utilizem livremente.

●**Tradução.** Traduzir páginas dos sites, instaladores, descrição de pacotes, documentação, notícias, alertas de segurança, entre outros. Tornar os sistemas usáveis por mais pessoas também tem o objetivo de atrair mais contribuidores para o projeto. Quando você traduz do Inglês para o Português por exemplo algum artefato, acredite, essa é uma excelente maneira de melhorar seus idiomas.

Participar de projetos de software livre e código aberto possibilita interagir com um público grande, sem exigências de renda. Sistemas oriundos desses projetos oferecem o requisito não-funcional da portabilidade. Este requisito trata da capacidade de trocar informações ou executar o software entre diferentes plataformas. Exemplo: Até poucos anos, o IRPF só podia ser declarado em computadores que rodavam *Microsoft Windows*. Hoje, pode ser declarado em quase todos os tipos de equipamento.

“Software Livre” e “Software de Código Aberto” são coisas diferentes

Vamos apresentar então brevemente a história de alguns elementos essenciais para entendermos melhor os conceitos de software livre e código aberto (*open source*). Em 1965 Ken Thompson e Dennis Ritchie (Bell labs, MIT, At&T) começaram a desenvolver o sistema operacional *Multics* (algo como múltiplos). Mas a junção dessas empresas não deu muito certo. Cada uma possuía interesses específicos e contraditórios. Em 1969 começaram a escrever um outro sistema operacional chamado Unics e depois renomeado para Unix. Escreveram usando a linguagem *Assembly*. Em 1973 reescreveram o Unix em linguagem C. Ken Thompson criou a linguagem B, que antecedeu a linguagem C. Para se ter uma ideia do quanto atual ele permanece, ele é um dos criadores da linguagem Go.

GNU é um sistema operacional tipo Unix criado por Richard Stallman, cujo objetivo desde sua concepção é oferecer um sistema operacional completo e totalmente composto por software livre. O nome “GNU” é um acrônimo recursivo para “GNU’s Not Unix!” (em português, é traduzido como “GNU Não é Unix!”). “GNU” é pronunciado como “menu”, com “g” em vez de “me”, e como o animal de origem africana “gnu”. O desenvolvimento do GNU, que começou em Janeiro de 1984, é conhecido como Projeto GNU.

Muitos dos programas em GNU são lançados sob responsabilidade do Projeto GNU (os pacotes GNU). Stallman também criou um mecanismo legal de garantia para que todos pudessem desfrutar dos direitos de copiar, redistribuir e modificar software, o que deu origem a *General Public License (GPL)*. E para institucionalizar o Projeto GNU, Stallman fundou a *Free Software Foundation (FSF)*. O software em um sistema semelhante ao Unix que aloca recursos da máquina e conversa com o hardware é chamado “kernel”. Linus Torvalds em 1991 começou o desenvolvimento do kernel Linux. GNU é tipicamente usado com um kernel Linux. Essa combinação é o Sistema Operacional GNU/Linux.

Software livre (*free software*) e código aberto (*open source*) são dois termos que geram muita confusão. Software livre é um movimento social, criado pela *FSF*, que visa a manutenção de quatro liberdades básicas de software para as pessoas. Trata-se de uma filosofia. Um software é dito livre se os usuários possuem as quatro liberdades essenciais:

● **A liberdade de executar o programa, para qualquer propósito (liberdade 0).**

● **A liberdade de estudar como o programa funciona, e adaptá-lo às suas necessidades (liberdade 1). Para tanto, acesso ao código-fonte é um pré-requisito.**

● **A liberdade de redistribuir cópias de modo que você possa ajudar ao próximo (liberdade 2).**

● **A liberdade de distribuir cópias de suas versões modificadas a outros (liberdade 3). Desta forma, você pode dar a toda comunidade a chance de beneficiar de suas mudanças. Para tanto, acesso ao código-fonte é um pré-requisito.**

Assim sendo, software livre é uma questão de liberdade, não de preço. Para entender o conceito, pense em “liberdade de expressão”, não em “cerveja grátis”.

Open source é uma metodologia de desenvolvimento, criada pela *Open Source Initiative (OSI)*, que respeita os mesmos princípios do software livre. Trata-se de uma visão mais técnica do que filosófica. De uma maneira geral, os dois termos descrevem quase a mesma categoria de software, porém eles referem-se a visões baseadas em valores fundamentalmente diferentes. Mas a visão da *FSF* e da *OSI* são diferentes em relação a algumas licenças. Assim sendo, há divergências. Muitas vezes trata-se de interpretação de texto. O conceito de *open source* veio diretamente da *Debian Free Software Guidelines (DFSG)*.

Open source são programas cujo código fonte é disponibilizado pelo Desenvolvedor. Mas isto não quer dizer que sejam considerados como software livre.

Existe o termo FOSS: *Free and Open Source Software*. Um programa *open source* pode não ser *FOSS*. Um programa *FOSS* pode não ser software livre. Em suma, para ser software livre, as quatro liberdades essenciais devem ser possíveis.

Freeware é um software disponibilizado para uso, sem a exigência de pagamento de licença de uso, ou com licença opcional. Não significa que o código fonte estará disponível, e, se estiver, não significa que você tenha as liberdades de alterá-lo e redistribuí-lo. Licenciamento de software sempre foi um tema polêmico ou nebuloso para muitos. A licença de uso é quem diz se é um software livre ou não. Software livre, geralmente, usa uma licença do tipo *CopyLeft*. Este tipo é uma maneira de usar a legislação de proteção dos direitos autorais com o objetivo de retirar barreiras à utilização, difusão e modificação de uma obra criativa devido à aplicação clássica das normas de propriedade intelectual, exigindo que as mesmas liberdades sejam preservadas em versões modificadas.