

# TLS: Libre Software

**Soluções em Software Livre**

# Ubiquidade da Computação



# Ubiquidade da computação

Em mais de meio século de evolução a computação digital *aprimorou-se exponencialmente* e encontrou uso em virtualmente *todos os setores* dos afazeres humanos.



#Transistors = 2,300  
Size = 29x12mm



#Transistors = 2.5+ billion  
Size = 53x45 mm



# Ubiquidade da computação

- *Setores públicos e privados*
- *Organizações de todo tipo*
- *Áreas militares e civis*
- *Institutos de pesquisa e educação*
- *Empresas de qualquer porte*

- *Bibliotecas*
- *Aeroportos*
- *Carros*

- *Indústria*
- *Comércio*
- *Bancos*
- *Escritórios*
- *Hospitais*

- *Casas*
- *Eletrodomésticos*
- *Bolsos e bolsas*

# Ubiquidade da computação

A computação é empregada para *toda finalidade concebível*.

- *Controle de processos*
- *Gestão de recursos*
- *Planejamento*
- *Tomada de decisão*
- *Simulação*

- *Automação*
- *Educação*
- *Organização*
- *Lazer*
- *Pesquisa*

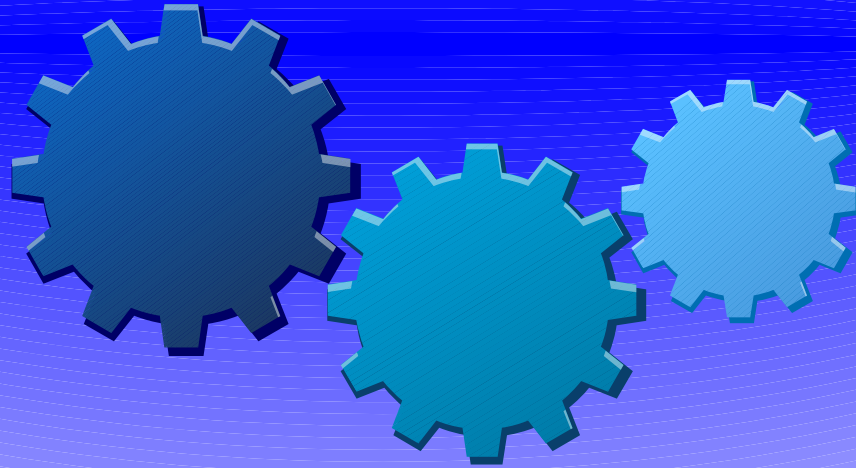
# Ubiquidade da computação

A ordem social depende da *eficiência* e do *funcionamento* dos sistemas de informação, pois eles compõem a *infraestrutura* que *sustenta* a sociedade humana.

- *Economia*
- *Produção de bens*
- *Transporte*

- *Comunicação*
- *Defesa*
- *Saúde*

# Software



# Software

Os dispositivos informáticos são em sua grande maioria *máquinas de computação geral*, que podem *solucionar qualquer tipo de problema* para os quais são programados.

- *Servidores*
- *Estações de trabalho*
- *Notebooks*
- *Tablets*

- *Celulares*
- *Videogames*
- *Dispositivos embarcados*



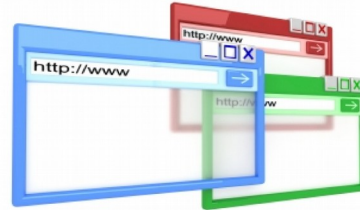
# Software

Em princípio o computador em si; o *hardware* ou *parte física*, *não é um fator determinante* nas potenciais soluções que o dispositivo pode oferecer para um problema em questão. Não que o hardware não seja importante, mas sim *secundário*.

Hardware



Software



# Software

Todos os sistemas de computação são regidos por software, que determinam a *funcionalidade*, *produtividade*, *usabilidade* e *confiabilidade* deles. Portanto, o *software* ou *parte lógica*, isto é, a *programação da máquina*, é o que define o quão útil e eficiente é uma solução informática em particular.

```
136 printf("[*] Android local root exploit (C) The Android Exploit Crew\n");
137
138 basedir = "/sqlite_stat_journals";
139 if (chdir(basedir) < 0) {
140     basedir = "/data/local/tap";
141     if (chdir(basedir) < 0)
142         basedir = strdup(getcwd(buf, sizeof(buf)));
143 }
144 printf("[+] Using basedir=%s, path=%s\n", basedir, path);
145 printf("[+] opening NETLINK_KOBJECT_UEVENT socket\n");
146
147 nenaset(&snl, 0, sizeof(snl));
148 snl.nl_pid = 1;
149 snl.nl_family = AF_NETLINK;
150
151 if ((sock = socket(PF_NETLINK, SOCK_DGRAM, NETLINK_KOBJECT_UEVENT)) < 0)
152     die("[~] socket");
153
154 close(creat("loading", 0666));
155 if ((ofd = creat("hotplug", 0644)) < 0)
156     die("[~] creat");
157 if (write(ofd, path, strlen(path)) < 0)
158     die("[~] write");
```

```
0001 0001 0000 0011 0000 0101 0101 0110 0111
1000 0000 0000 0011 0000 0101 0101 0101 0101
0110 0111 1000 0001 0001 0000 0011 0100 0100
0101 0110 0111 0000 0000 0000 0000 0100 0101
0110 0111 1000 0000 0000 0011 0000 0101 0101
0101 0101 0110 0111 0000 0000 0000 0011 0100
0100 0101 0000 0111 0000 0000 0000 0011 0000
0000 0000 0000 0000 0001 0001 0001 0010 0011
0011 0100 0101 0110 0111 1000 0001 0001 0100
0011 0100 0101 0110 0111 0000 0000 0000 0011
0011 0011 0100 0101 0110 0111 0000 0000 0000
0011 0000 0101 0101 0101 0101 0110 1000
0000 0000 0011 0011 0000 0101 0101 0101 0101
0110 0111 0000 0000 0000 0000 0000 0101 0101
0101 0101 0110 0111 1000 0000 0000 0000 0100
0000 0110 0111 0000
```

# Software

Um *mesmo programa* de computador pode ser executado numa *ampla variedade de computadores*, de tipos, tamanhos, recursos e gerações diferentes.



*Privativo*



# Software Privativo

*Tradicionalmente* empresas especializadas em construção de software produzem soluções informáticas para os diversos setores do mercado e sociedade segundo um *modelo privativo*.



# Software Privativo

Neste modelo a empresa se usa da *lei de copyright* para licenciar o software sob *direitos legais exclusivos* que permite ao cliente licenciado usar o programa apenas sob certas condições, *restringindo o uso* e *proibindo a modificação, compartilhamento, estudo, redistribuição* e até mesmo a prática de *engenharia reversa*.



copyright

all rights reserved

# Software Privativo

O cliente *não compra o programa*, mas sim uma *licença de uso* que normalmente permite a utilização em *apenas uma máquina* e não raramente somente para *propósitos pré-especificados* pelo fabricante. É comum também que a licença tenha *prazo de validade*.





# Software Privativo

Adicionalmente ao *copyright*, as empresas desenvolvedoras de software podem se usar de *contratos*, *patentes* e *segredos comerciais* como base legal para obter *direitos exclusivos* sobre os programas que produzirem.



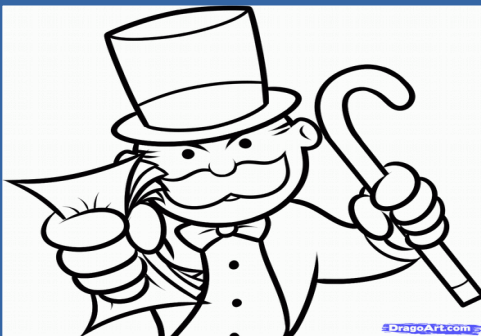


*Desvantagens*



# Desvantagens

*Submissão ao monopólio da empresa desenvolvedora.* Só ela pode oferecer atualizações, correções, suporte e serviço, e ela tem *inquestionável autoridade* com respeito às funcionalidades, segurança e futuro do programa. Se for do interesse do desenvolvedor *não resolver uma falha*, ou *descontinuar o produto*, ou mesmo se ele vier à *falência*, *não há nada que o cliente possa fazer.*



# Desvantagens

*Perda do controle da sua própria computação.* O cliente *não tem autonomia* para inspecionar a correção e confiabilidade dos métodos do programa ou modificá-los para atender às suas necessidades particulares. O código fonte do programa é tido como *segredo comercial* e *ninguém exceto o desenvolvedor original* tem acesso e direito sobre ele.



No Ctrl

# Desvantagens

*Impotência frente a imposições arbitrárias.* O desenvolvedor do programa pode *decidir fazer qualquer tipo de alteração* que julgar apropriada na próxima versão do programa, e *impor a atualização* por meios técnicos àqueles que *dependem inexoravelmente do seu produto*.



# Desvantagens

*Guerra ao compartilhamento e colaboração.* O compartilhamento do programa é *tido como crime* e o cliente fica *impossibilitado de colaborar* com parceiros que possam ter necessidades e objetivos similares.



# Desvantagens

*Dependência forjada a uma tecnologia secreta e a um único fornecedor.*  
O desenvolvedor do software usa de *meios técnicos e legais inoportunos* ao cliente para garantir sua *exclusividade e aparente importância*.

DUBIOUS TECH PRACTICES:

A HISTORY  
OF VENDOR  
LOCK-IN

A padlock is shown against an orange background. The padlock's body is covered in binary code (0s and 1s). The text 'DUBIOUS TECH PRACTICES:' is in a dark grey box above the padlock, and 'A HISTORY OF VENDOR LOCK-IN' is written in white over the padlock.

# Desvantagens

*Escassez artificial.* Apesar de a tecnologia digital permitir o desenvolvimento de um programa uma única vez e sua conseguinte *aplicação e exata replicação infinita*, os desenvolvedores de programas privativos *geram uma situação de escassez artificial*, fazendo com que programas de computador, que são *entidades virtuais*, sejam *entendidos e tratados como bens físicos e finitos*.





# Desvantagens

*Obsolescência planejada.* Software privativo é lançado com o *fim da sua vida útil já planejada*. Tal planejamento se baseia unicamente nos *interesses comerciais dos desenvolvedores* e não necessariamente nas necessidades do cliente que é *compelido por razões técnicas* a atualizar seus sistemas *contra sua comodidade ou vontade*.





# Desvantagens

*Alto Custo.* Desenvolvedores de software privativo geralmente usam *práticas abusivas* de licenciamento de software, com *preços tão altos* quanto *menos concorrência e mais dependência* de seu produto existir, com recorrente renovação de licenças e licenciando por instância e funcionalidade do programa.



# Desvantagens

*Anti-recursos.* Software privativo comumente é dotado de *anti-recursos* que colocam o cliente numa *posição desvantajosa* com relação ao desenvolvedor e ao controle de sua própria informática.

- *Limitada interoperabilidade;*
- *Uso de formatos secretos;*
- *Retro-incompatibilidade deliberada;*
- *Limitações arbitrárias;*
- *Segurança pela obscuridade;*

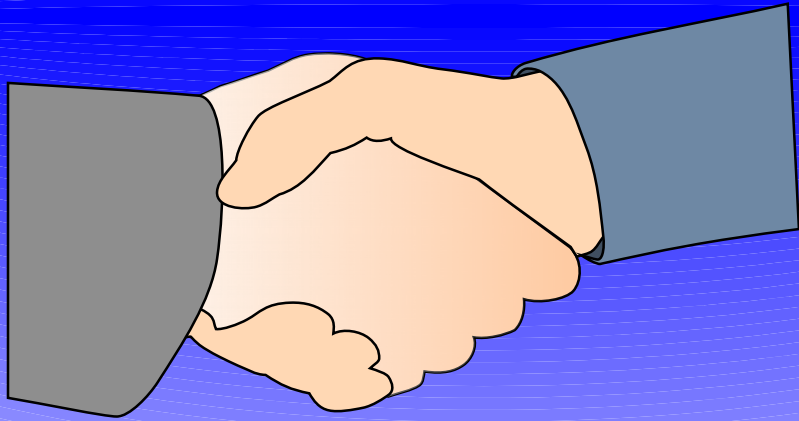
# Desvantagens

**Malware.** Mais comumente do que se possa esperar, software privativo inclui *funcionalidades maliciosas* inseridas pelos próprios desenvolvedores com a finalidade de *obter vantagem comercial* ou *satisfazer requisições governamentais*.

- *Back doors;*
- *Censura;*
- *Insegurança;*
- *Sabotagem;*

- *Vigilância;*
- *DRM;*
- *Prisões;*
- *Tiranos;*

**Livvre**



# Software Livre

*Software livre* é todo o software que *não é privativo*, isto é, cujo detentor dos direitos autorais provê a qualquer usuário as *liberdades necessárias* para *o controle de sua própria informática*.

- *Uso do programa para qualquer propósito*
- *Estudo e modificação do programa sem restrições*
- *Livre redistribuição do programa*
- *Redistribuição das modificações feitas ao programa*

# Software Livre

O *software livre* dá *autonomia* ao usuário pois seus termos de licenciamento exigem a provisão do código fonte e a sustentação das *liberdades essenciais* de *execução*, *cópia*, *distribuição*, *estudo*, *mudança* e *aprimoramento*.



***Free as in Freedom***

# Software Livre

Software livre é uma questão *de liberdade, não de preço.*



# Freedom

**Vantagens**





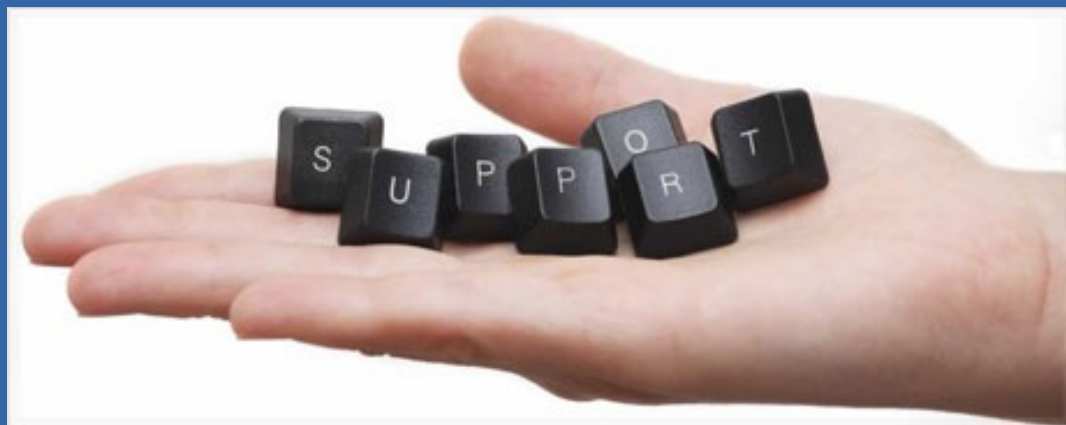
# Vantagens

*Um mercado livre de monopólios.* O cliente tem o *direito de fazer melhorias* no programa sem ter que se submeter a um fabricante em particular. *Não há mais o risco* de um produto de software ser descontinuado, pois qualquer pessoa, entidade ou organização pode *manter uma versão* do programa *para seus próprios propósitos*.



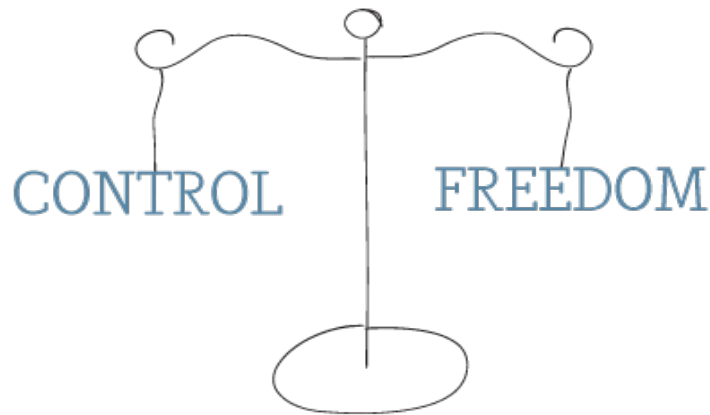
# Vantagens

*Um mercado livre para a prestação de serviços.* Qualquer pessoa ou empresa com a *qualificação técnica adequada* pode prestar serviços de suporte e desenvolvimento ou customização do programa. *O cliente tem escolha.*



# Vantagens

*Liberdade e controle.* O cliente pode *inspecionar e modificar* o programa *quando quiser e como quiser* para atender *suas próprias necessidades*, pois ele tem o *código fonte* do programa e os *direitos legais* para fazê-lo.



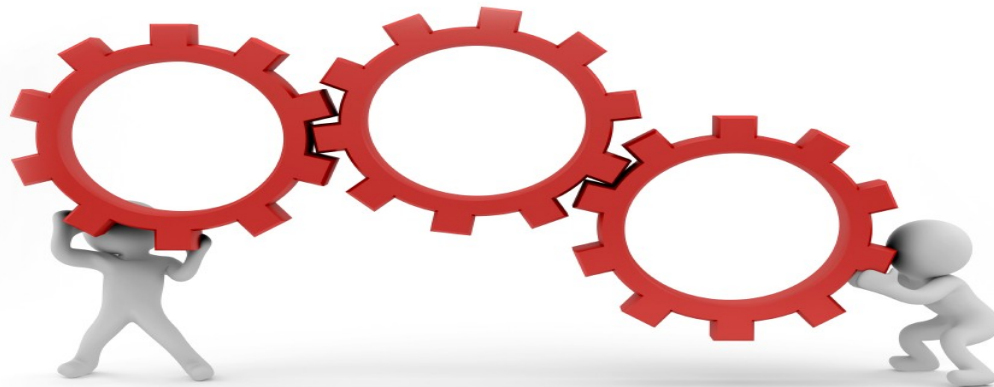
# Vantagens

*Poder de decisão.* O programa recebe as *funcionalidades que são úteis* ao cliente, que não precisa lidar com modificações arbitrárias feitas por um único fornecedor.



# Vantagens

*Incentivo ao compartilhamento e colaboração.* O cliente pode se *associar a parceiros* que compartilham de interesses e objetivos para *colaborar* na construção de uma *solução conjunta* baseada no programa.



# Vantagens

*Abundância natural.* O programa pode ser *copiado*, *redistribuído* e usado numa *quantidade qualquer* de computadores, *sem prazo de validade* e para *qualquer propósito*.



# Vantagens

*Prazo de utilidade indefinido.* O programa *não fica obsoleto ou defasado* por causa de uma nova versão ou decisão de um desenvolvedor em particular, ele é *moderno e atual* enquanto for *suficiente aos propósitos* do cliente.



# Vantagens

*Sem custo de licenciamento.* Não é necessário pagar pelo licenciamento, por cópias ou uso de funcionalidades. O *único custo* será com a *manutenção* e *adaptação* do sistema.

**GRATIS**



# Desvantagens

*Livre de anti-recursos.* O software livre recebe apenas os *recursos* que os usuários *querem* e *precisam*.

- *Interoperabilidade*
- *Uso de formatos padronizados internacionalmente*
- *Retro-compatibilidade permanente*
- *Sem limitações arbitrárias*
- *Segurança por métodos validados matematicamente*

# Vantagens

*Livre de funcionalidades maliciosas.* Os usuários não desejam ser restringidos, aprisionados, censurados, sabotados ou vigiados, portanto *nenhuma dessas funcionalidades maliciosas* encontra um caminho para o software livre.



**GNU e FSF**



**FSF** FREE SOFTWARE  
FOUNDATION

# Projeto GNU e FSF

Em **1983** é lançado o *movimento do software livre*; um *movimento social* com o objetivo de *garantir as liberdades essenciais* dos usuários de programas de computador *promovendo o estudo, distribuição, desenvolvimento e modificação* universais de software.



# Projeto GNU e FSF

Em **1984** é fundado o **projeto GNU** para o desenvolvimento de um **sistema operacional** completamente livre chamado "**GNU**".



**GNU**  
run free run GNU  
.....

# Projeto GNU e FSF

Em **1985** é iniciada a *Fundação para o Software Livre (FSF)*, para *apoiar* e *avançar* o movimento do software livre. Ela é a *principal entidade patrocinadora* do *projeto GNU* e *referência mundial* no assunto.

The logo for the Free Software Foundation (FSF) is rendered in a pixelated, binary style. The letters 'F', 'S', and 'F' are the central focus, with the 'S' being significantly larger and more prominent. The entire logo is set against a background of binary code (0s and 1s) and is enclosed within a thick red rectangular border.

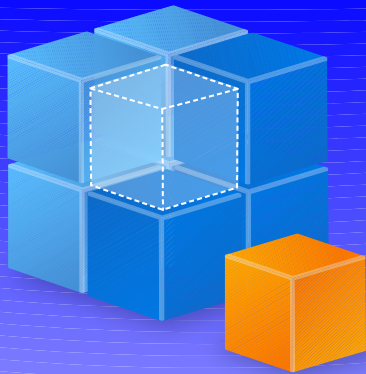
0101010101010101  
101 **F** 101 **S** 101 **F** 1010  
010 0101010101010101  
101010101010101010

# Projeto GNU e FSF

Hoje é possível se usar sistemas de computador *rodando apenas software livre* por meio do *sistema operacional GNU*, comumente usado com o *kernel* chamado "*Linux*". Milhões de pessoas o fazem.



# *Modelo de Desenvolvimento*





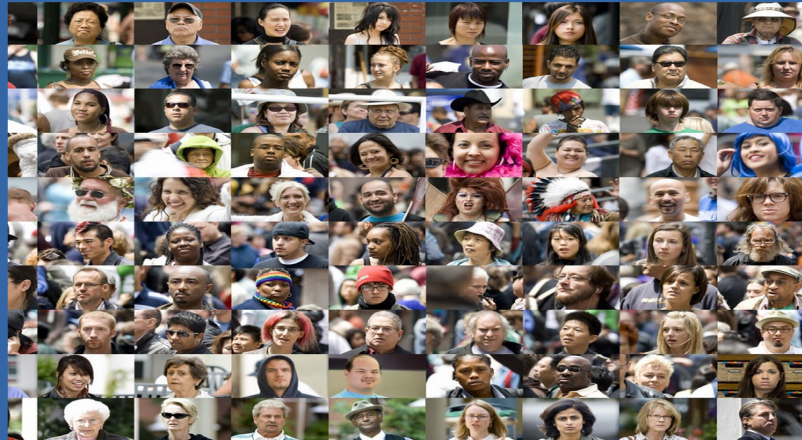
# Modelo de Desenvolvimento

*Software livre* é geralmente desenvolvido de *maneira distribuída* pela *internet*.



# Modelo de Desenvolvimento

De poucas a *milhares de pessoas*, de *todos os continentes*, contribuem para o desenvolvimento de um pacote de software livre no qual estejam interessadas. *Qualquer pessoa* com *interesse e habilidade técnica* pode contribuir.



# Modelo de Desenvolvimento

*Programadores* e *usuários* de software livre *colaboram* juntos com o *desenvolvimento* do programa.

- *Código*
- *Documentação*
- *Apontamento de falhas*

- *Traduções*
- *Teste*
- *Divulgação*

# Modelo de Desenvolvimento

Não só indivíduos mas *empresas, organizações, entidades governamentais* e de *pesquisa* estão *envolvidas no desenvolvimento* de software livre.

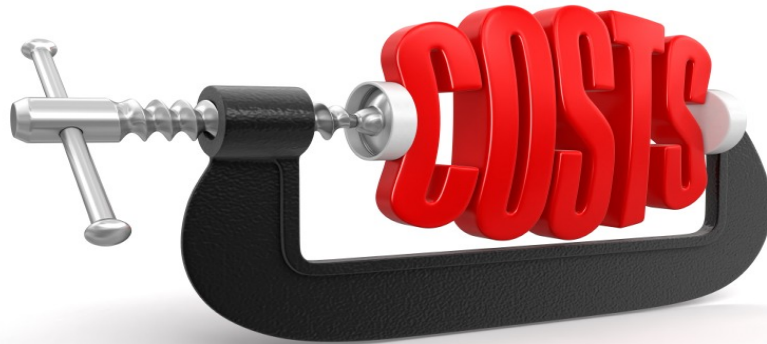


**Vantagens**



# Vantagens

*Redução de custo.* O desenvolvimento muitas vezes acontece de *maneira voluntária* por *partes interessadas* que se *beneficiam mutuamente* partilhando do produto final, sem a necessidade de investimento em times dedicados de desenvolvimento.



# Vantagens

*Boa penetração no mercado.* Software livre comumente se estabelece como um *padrão industrial*, ganhando *vantagem competitiva* e a *lealdade dos desenvolvedores* que tem a sensação de estarem no controle.



International  
Organization for  
Standardization

# Vantagens

*Rapidez no desenvolvimento.* Um grande projeto de software livre é rapidamente construído por centenas de *programadores contribuintes*, dada sua *natureza distribuída*.





# Vantagens

*Qualidade e robustez.* Com o *código publicamente disponível* e a grande quantidade de olhos que o observam, todo tipo de *erro se torna trivial* e *rapidamente corrigível*.



# Vantagens

*Generalidade de aplicação.* Comumente *diversos grupos* de *diferentes realidades*, e *partes do planeta*, participam do desenvolvimento de um mesmo pacote de software, fazendo com que ele solucione tipos de problemas que não foram antecipados pelo *planejamento inicial*.



# Vantagens

*Eficiência e correção.* Dado que os *lançamentos* se dão numa base *frequente* e são *extensivamente testados* em diversos ambientes, configurações e casos de usos distintos, o pacote de software livre se torna *eficiente e correto* numa fração do tempo tradicional.



# *Proposta*



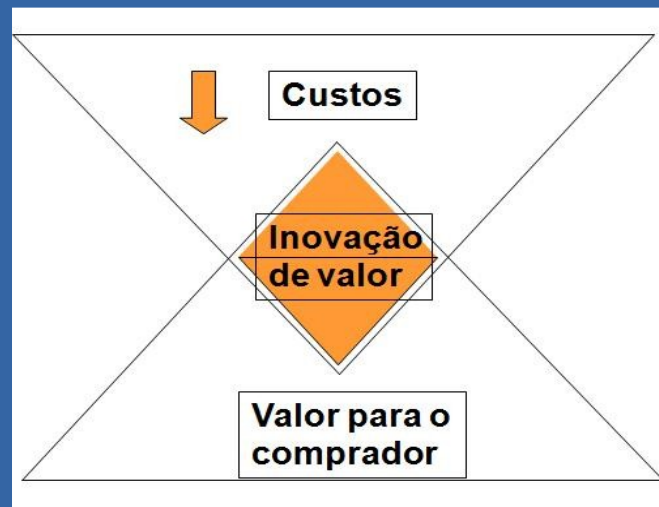
# Proposta

As *vantagens do software livre* e as *desvantagens do software privativo* como uma *estratégia de marketing*.



# Proposta

O *modelo de desenvolvimento* do software livre como uma *vantagem comercial*.



# Proposta

Construir a *reputação* da empresa e *consolidar nosso modelo* com base na nossa *experiência técnica*, *organizacional* e *processual* dentro do movimento de software livre.



# Quem Somos



***Allisson Antonio Alvarenga Garcia:***

Graduando em ciência da computação – UFU;

Estagiário na empresa Martins Comércio e Serviço S/A;

Aluno de Pesquisa (CNPQ) em Mineração de Dados;



***Bruno Félix Rezende Ribeiro:*** um programador de computadores, administrador de sistemas, hacker GNU e ativista do software livre. Ele é reconhecido pela FSF como um membro do projeto GNU, onde trabalha com diversos grupos de especialistas em computação no desenvolvimento colaborativo e distribuído de software livre pela internet.



# Links

- Projeto GNU: <http://www.gnu.org/>
- Fundação para o Software Livre: <http://www.fsf.org/>
- Portal do Software Livre do Governo Brasileiro:  
<http://www.softwarelivre.gov.br/>
- SERPRO: <https://www.serpro.gov.br/>