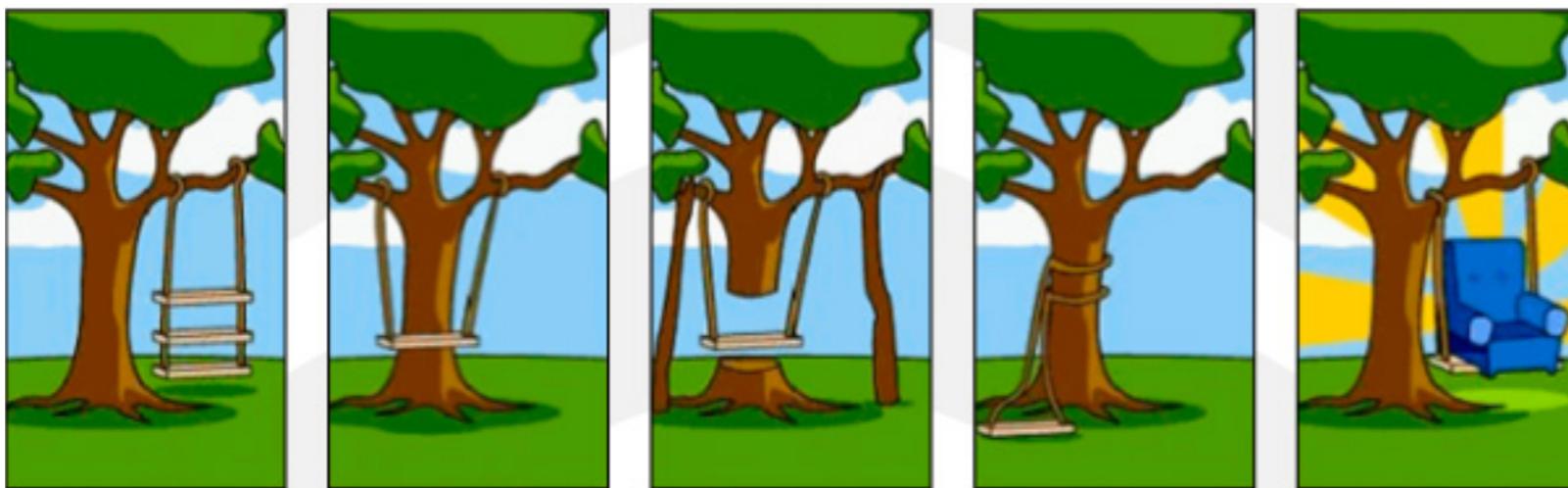


Tópico Especial 04

“Vilões” do Código

Programar É Fácil
Desenvolver Software É Difícil



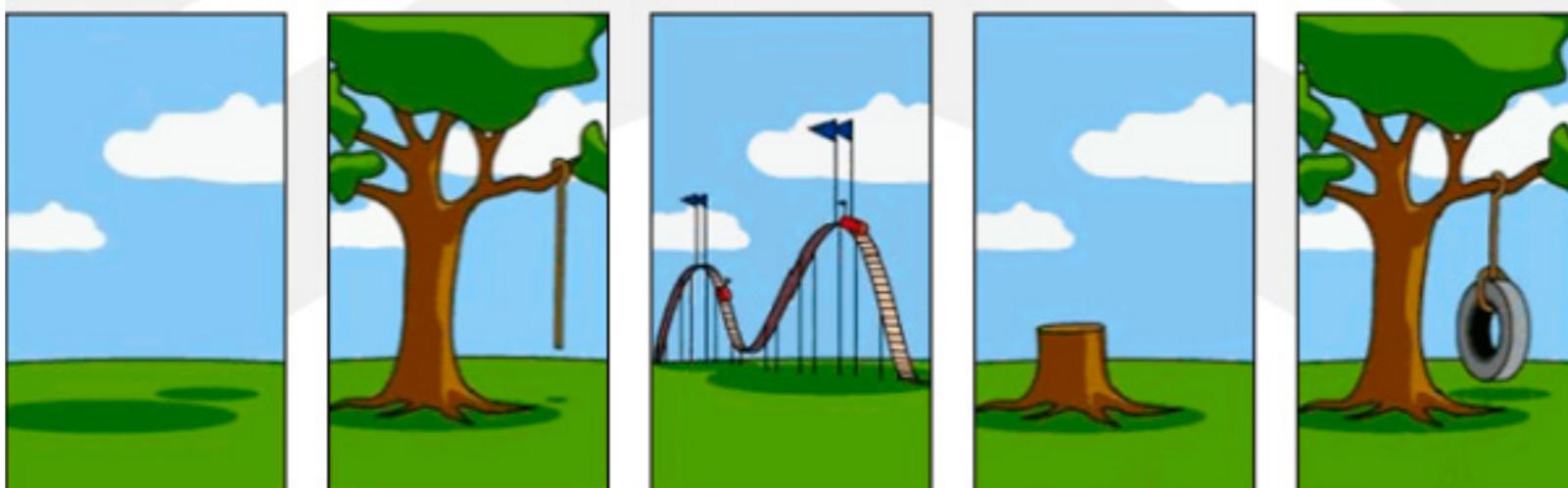
Como o cliente
explicou

Como o gerente
do projeto
entendeu

Como o
engenheiro
desenhou

Como o
programador
escreveu

Como o vendedor
descreveu



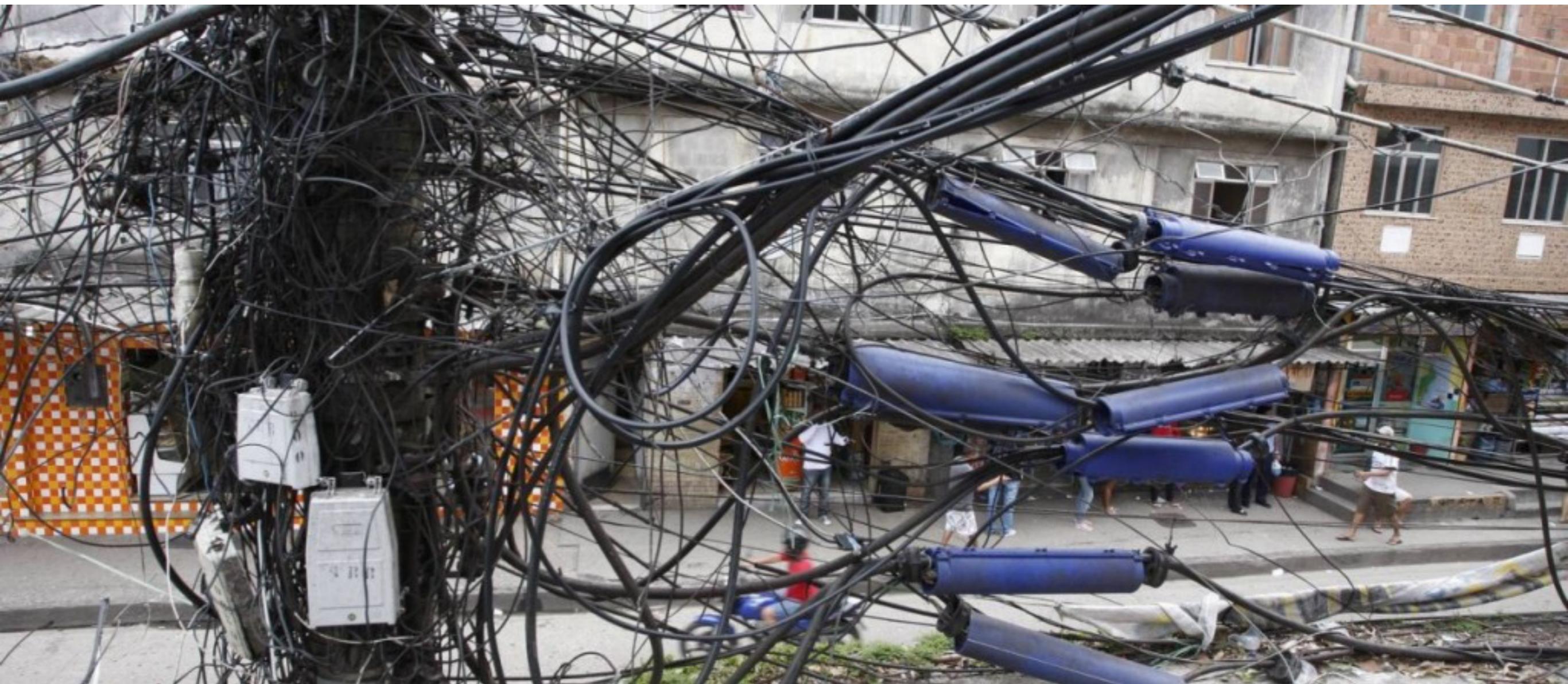
Como o projeto foi
documentado

Como o projeto foi
executado

Como o cliente foi
cobrado

Como o suporte foi
prestado

Do que o cliente
precisava



desciclopedia.org/wiki/Gambi_Design_Patterns

Gambi Design Patterns - Desciclopédia

Crie uma conta Entrar

Artigo Discussão Ler Editar Ver histórico Pesquisa

Temos 53 880 artigos! Ajude-nos a crescer ainda mais criando o seu artigo!

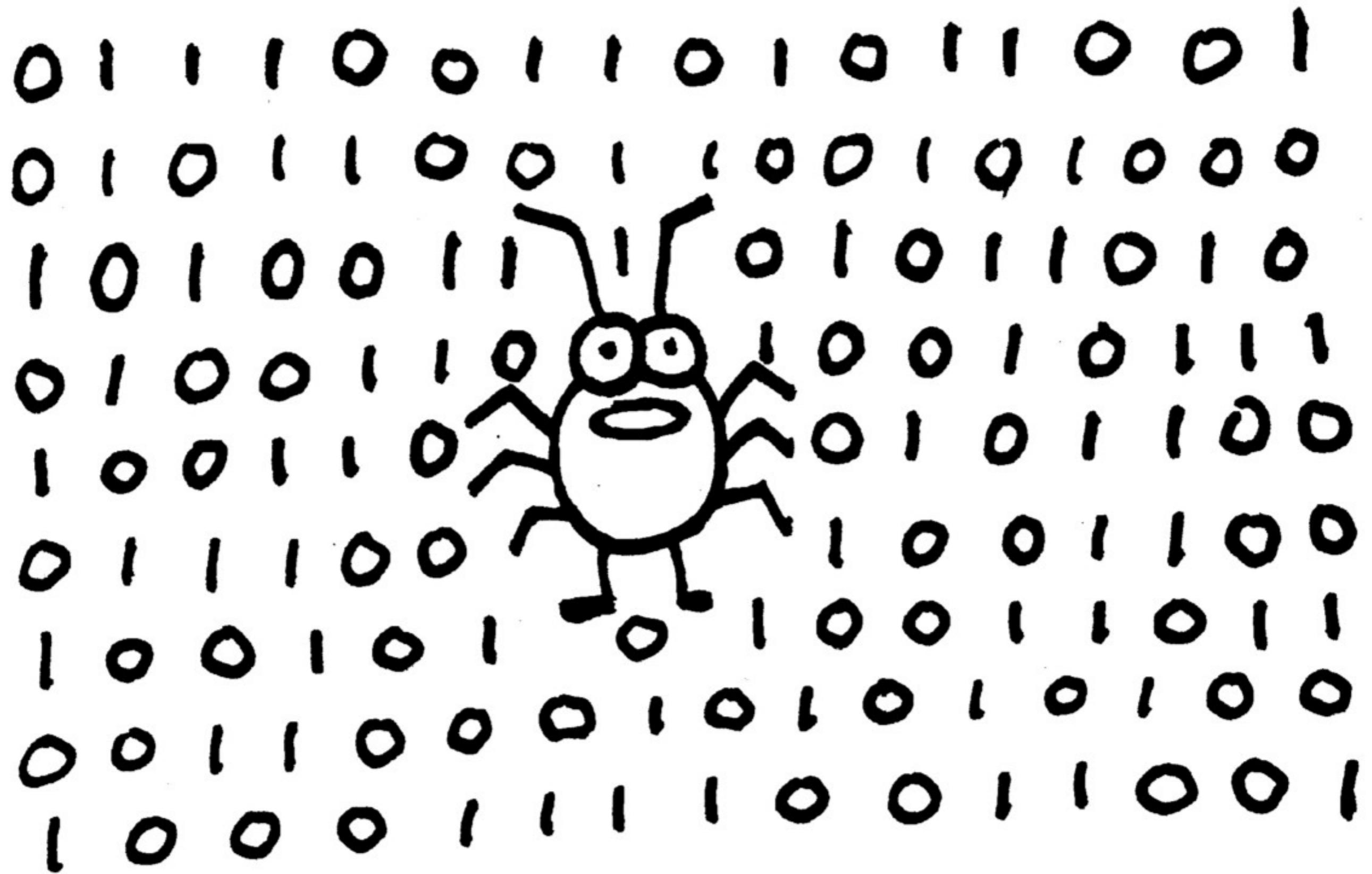
(Problemas para acessar desciclopedia.org? Utilize nosso domínio alternativo [.](#))

Gambi Design Patterns

Atenção: Este artigo, de forma semelhante aos programas que ele descreve, cresceu e sofreu modificações de forma desordenada e desorganizada, por autoria de diversas pessoas. O resultado é que este artigo é excessivamente longo, pesado e desestruturado, exatamente como qualquer programa que segue a metodologia POG. Este artigo precisa ser refatorado em artigos menores, mas nenhum dos POGramadores que são autores dele têm tempo, vontade ou cérebro o suficiente para fazer isto, preferindo então seguir os conceitos da POG e ficar apenas remendando-o e inchando-o cada vez mais.

As Soluções Técnicas de Contorno (STC), também chamadas de Engenharias de Emergência (EE), Artifícios Técnicos de Emergência (ATE) ou Solução Criativa (aka. gambiarras) podem ser aplicadas de várias formas. No entanto, não há ainda consenso entre desenvolvedores e analistas acerca de padrões de projeto para desenvolvimento orientado a gambiarras. Uma prática muito comum é criação de uma entidade específica, com o intuito único de prover uma funcionalidade emergencial a um sistema. Não são raros os casos de utilização de classes "`gambiarra.java`", "`agoravai.java`", "`funcionapeloamordedeus.cpp`", métodos "`void ajeita()`", "`public String bruto()`" e afins. Em alguns casos, métodos com nome sofisticados como "`setMagicOn()`" e "`setApellationMode(true)`" também são utilizados para dar um ar mais profissional. Também para tornar o código mais amigável e legível alguns programadores fazem uso abusivo do MO de métodos, utilizando nomes como "`check()`", "`checka()`", "`yeka()`" ou "`pussu()`". São os chamados **Gambiwares**, componentes de software que promovem a

011100110101011001
01011001100100101000
101001110101011010
010011000100101111
10011001010101100
01110011001001100
10010101010011011
00110000101010100
1000111100110011001







SUPER VILÕES DO CÓDIGO

VOLUME 1



CHARADA



**DESVENDE O *NOME* DAS
MINHAS *VARIAVEIS!***

ESTILO EQUAÇÃO

```
x = readtable('t.xlsx');
v = x{:, 1} * 100;
t = x{:, 2};
a = 2000 : 2010;
plot(a, v);
```

ESTILO ICQ

```
tab = readtable('xxx.xlsx');
vals = tab{:, 1} * 100;
tot = tab{:, 2};
dat = 2000 : 2010;
plot(dat, v);
```

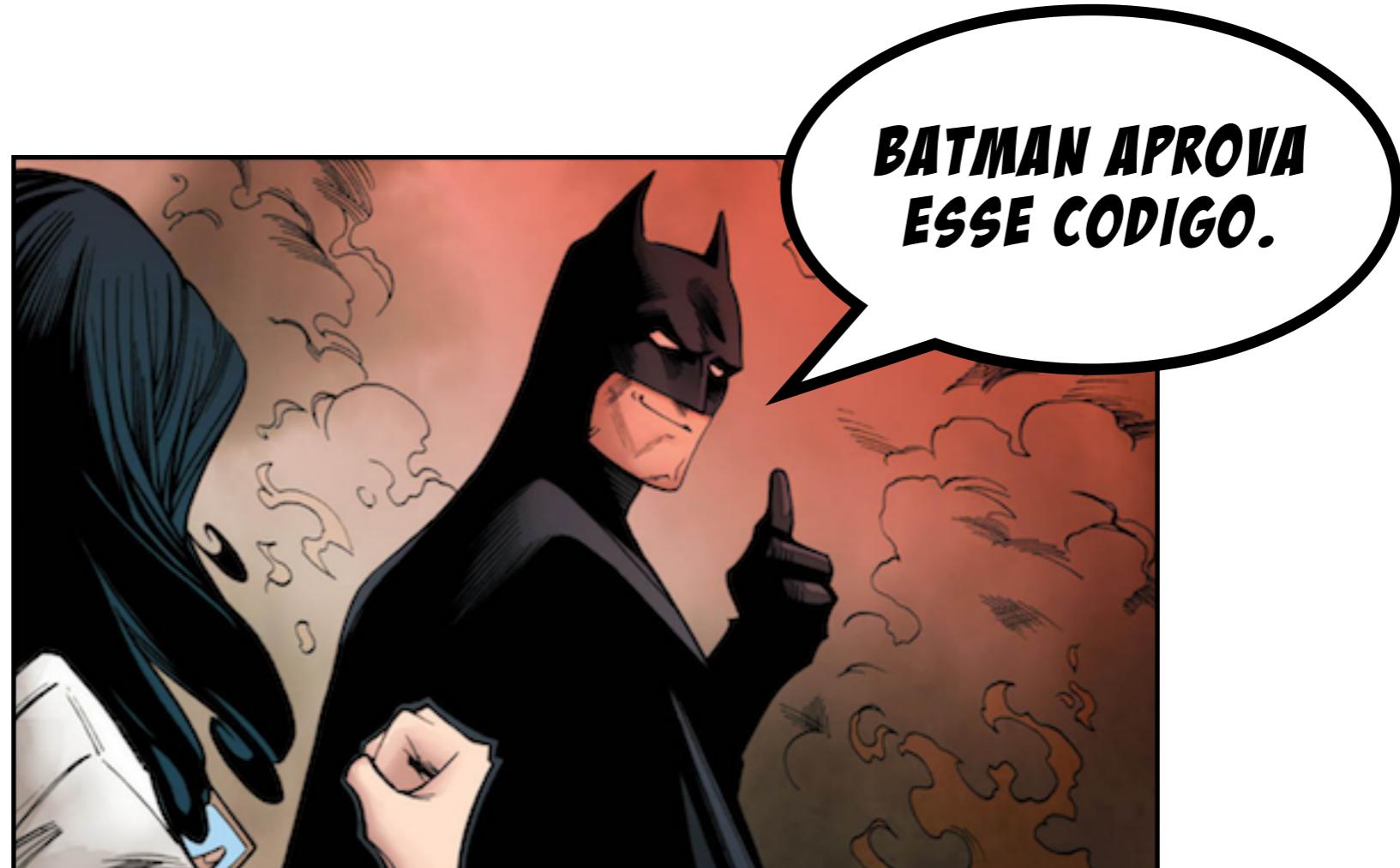
ESTILO CORINGA

```
boo = readtable('caos.xlsx');
torta = boo{:, 1} * 100;
surpresa = boo{:, 2};
hahaha = 2000 : 2010;
plot(hahaha, torta);
```



ESTILO LEGÍVEL

```
dadosFinanceiros = xlsread('dados.xlsx');
rendimentos = dadosFinanceiros{:, 1} * 100;
custos = dadosFinanceiros{:, 2};
anos = 2000 : 2010;
plot(anos, rendimentos);
```



COMENTÁRIOS INÚTEIS

```
% carrega dados da tabela  
dados = readtable('dados.xlsx');  
% multiplica valores por mil  
vals = dados{:, 1} * 1000;  
% plota valores  
plot(a, vals);
```



COMENTÁRIOS ÚTEIS

```
dados = xlsread('dados.xlsx');  
  
% a planilha está em quilômetros,  
% mas a fórmula usa metros  
distancias = x(1, :) * 1000;  
  
plot(anos, rendimentos);
```

**MAS DA TRABALHO ESCREVER
TANTO TEXTO E...**

**DA MUITO MAIS TRABALHO
LER SEM ISSO!**



MAGNETO



**TUDO GRUDADO
MAGNETICAMENTE!**

ESTILO “PROIBIDO USAR ESPAÇOS”

```
x=0:0.001:150;  
y=sqrt(sin(2*pi*x)+log(x)/100)-89.9;  
plot(x(1:2:end),y(1:2:end),'b.-');  
title('Sinais'); xlabel('x'); ylabel('y');
```

ESTILO “ECONOMIA DE LINHAS”

```
y=sum(sum(matriz(12:100,45:78)),2)-mean(vetor(x-12:end))+sin(0:0.001:pi);
```

ESTILO “TEXTÃO”

```
valoresAnoAAño = valoresDeMercado(:, 1 : 12 : end);  
variaciones = diff(valoresAnoAAño, 1, 2);  
percentuaisDeCrescimento=variaciones./valoresDeMercado(:, 1:12:end-12);  
subplot(2, 2, [3 4]);  
bar(2008 : 2017, percentuaisDeCrescimento');  
title('Crescimento Ano a Ano');  
legend('Apple', 'Facebook', 'Google', 'Microsoft');  
xlim([2007 2018]);  
xlabel('Ano');  
ylabel('Crescimento (%)');
```

ESTILO “PARÁGRAFOS DE CÓDIGO”

```
valoresAnoAAño = valoresDeMercado(:, 1 : 12 : end);
variaciones = diff(valoresAnoAAño, 1, 2);
percentualesDeCrecimiento = variaciones ./ ...
    valoresDeMercado(:, 1 : 12 : end-12) * 100;

subplot(2, 2, [3 4]);
bar(2008 : 2017, percentualesDeCrecimiento');

title('Crecimiento Ano a Ano');
legend('Apple', 'Facebook', 'Google', 'Microsoft');
xlim([2007 2018]);
xlabel('Ano');
ylabel('Crecimiento (%)');
```

processamento
de dados

plotagem

identificações
do gráfico

CLONES

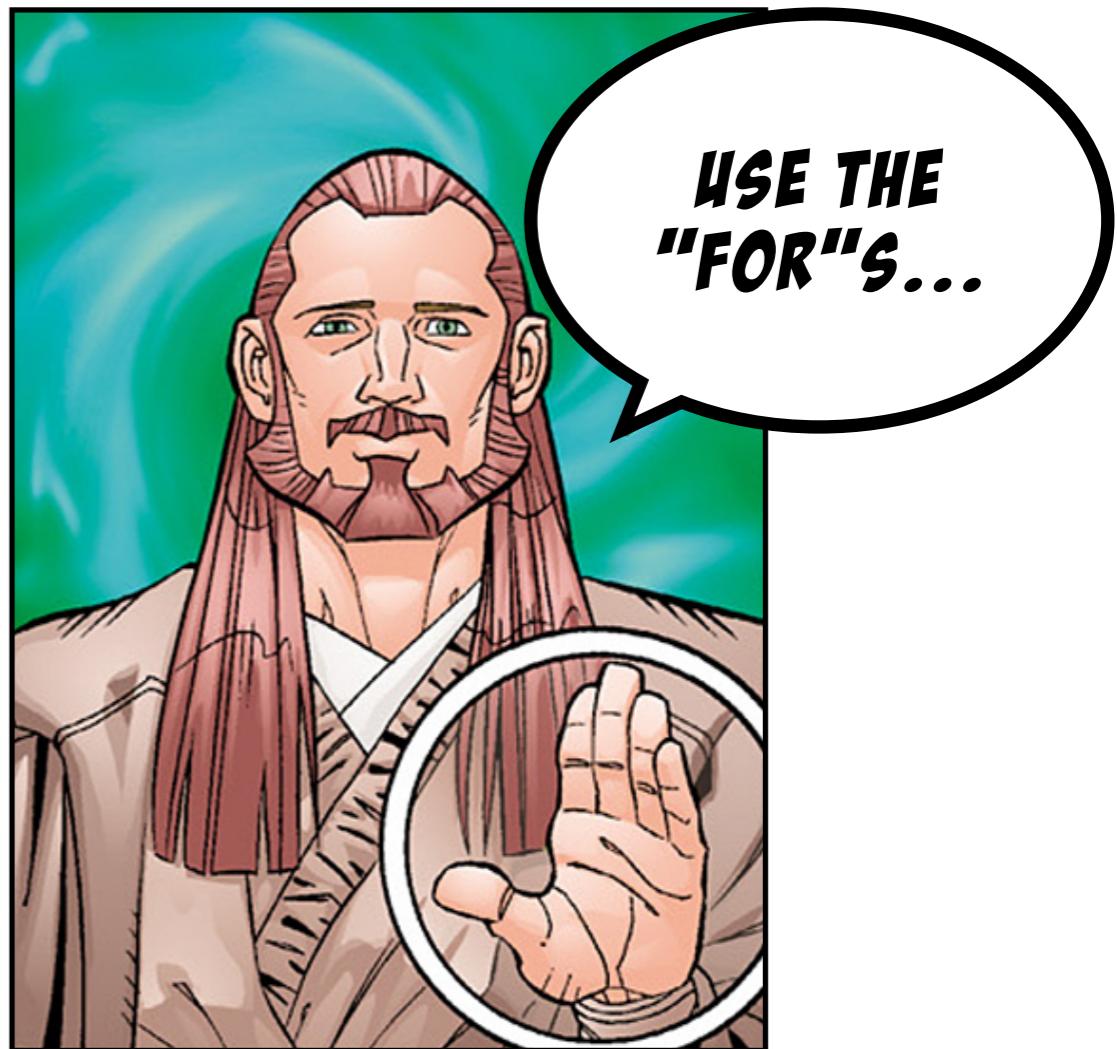
**ESTAMOS REPLICADOS
EM TODO O CODIGO!**



THE ENEMY WITHIN

CÓDIGO REPLICADO

```
x = 0 : 0.003 : 2;  
hold on;  
y = sin(2 * pi * 1 * x);  
plot(x, y);  
y = sin(2 * pi * 2 * x);  
plot(x, y);  
y = sin(2 * pi * 3 * x );  
plot(x, y);  
y = sin(2 * pi * 4 * x );  
plot(x, y);  
y = sin(2 * pi * 5 * x );  
plot(x, y);
```



CÓDIGO UNIFICADO

```
x = 0 : 0.003 : 2;  
hold on;  
for f = 1 : 5  
    y = sin(2*pi * f * x);  
    plot(x, y);  
end
```

CÓDIGO COMPLICADO

```
primeiroNegativo = [];
for i = 1 : length(vetor)
    if vetor(i) < 0
        primeiroNegativo = vetor(i);
        break;
    end
end
```

CÓDIGO SIMPLES

```
posicao = find(vetor < 0, 1);
primeiroNegativo = vetor(posicao);
```



TRECHO DE CÓDIGO 1

```
amostras = amostras / 1000;  
amostras( isnan(amostras) ) = 0;  
media = mean(amostras);  
variancia = var(amostras);
```

TRECHO DE CÓDIGO 2

```
sinal = sinal / 1000;  
sinal( isnan(sinal) ) = 0;  
media = mean(sinal);  
variancia = var(sinal);
```

TRECHO DE CÓDIGO 3

```
dados = dados / 1000;  
dados( isnan(dados) ) = 0;  
media = mean(dados);  
variancia = var(dados);
```

FUNÇÃO

```
function [media, variancia] = ObterEstatisticas(dados)  
  
dados = dados / 1000;  
dados( isnan(dados) ) = 0;  
media = mean(dados);  
variancia = var(dados);
```

**A FUNÇÃO NOS ENVOLVE E
PENETRA. EH O QUE MANTEM O
CÓDIGO UNIDO!**

TRECHO DE CÓDIGO 1

```
[media, variancia] = ObterEstatisticas(amostras);
```

TRECHO DE CÓDIGO 2

```
[media, variancia] = ObterEstatisticas(sinal);
```

TRECHO DE CÓDIGO 3

```
[media, variancia] = ObterEstatisticas(dados);
```



VOLDEMORT

**DOMINAREI O MUNDO COM
NUMEROS MAGICOS!**



CÓDIGO COM NÚMEROS MÁGICOS

```
amostragem = sinal(1 : 2 : 120485);

for i = 1 : 15657
    ...
end

dados = tabela{:, [5 13]}

valoresDeMercado = matriz(3, :);
```

LUMUS!



CÓDIGO SEM NÚMEROS MÁGICOS

```
amostragem = sinal(1 : 2 : end)

for i = 1 : length(vetor)
    ...
end

dados = tabela{:, {'Idades' 'Sexos'}}

LINHA_DA_APPLE = 3;
valoresDeMercado = matriz(LINHA_DA_APPLE, :);
```

SAURON



**UM ARQUIVO PARA A
TODOS GOVERNAR!**

```

escudoTime1 = strcat('escudo',Time1,'.png');
escudoTime2 = strcat('escudo',Time2,'.png');
imshow(escudoTime1, 'Parent', eixos2);
imshow(escudoTime2, 'Parent', eixos3);
cores={'r' 'k' 'g' 'b'};
coresTimes={'Flamengo' 'Botafogo' 'Fluminense' 'Vasco'};
indice_cor1=find(strcmp(coresTimes,Time1)==1);
indice_cor2=find(strcmp(coresTimes,Time2)==1);
%Garante se os times são diferentes ou indica que o comando foi inválido
if strcmp(Time1,Time2)==0
    times = {Time1 Time2};
    timesOrdenados = sort(times);
    nomeDaPlanilha = [timesOrdenados{1} 'x' timesOrdenados{2}];
    [num, text, tudo] = xlsread('ConfrontoGeral2.xlsx', nomeDaPlanilha);
    ColunaTime1=find(strcmp(text(1,:),Time1)==1)
    ColunaTime2=find(strcmp(text(1,:),Time2)==1)

[num_retrospecto, texto_retrospecto, tudo_retrospecto] = xlsread('RetrospectoBrasileiro.xlsx');
ColunaTime1Retro=find(strcmp(texto_retrospecto(1,:),Time1)==1);
ColunaTime2Retro=find(strcmp(texto_retrospecto(1,:),Time2)==1);

[num_torcida, texto_torcida,todo_torcida]=xlsread('MediasDePublico.xlsx');

if Tipo_de_grafico==1
    cla(eixos, 'reset');
    soma_de_gols_Time1= cumtrapz(num(:,ColunaTime1(1)));
    soma_de_gols_Time2= cumtrapz(num(:,ColunaTime2(1)));
    plot(eixos,num(:,2),soma_de_gols_Time1,cores{indice_cor1}, num(:,2), soma_de_gols_Time2, cores{indice_cor2});
    xlabel(eixos, 'Ano');
    ylabel(eixos, 'Gols');
    title(eixos, 'Histórico de Gols');
    legenda = legend(eixos, Time1, Time2);
    legenda.Location = 'southeast';

elseif Tipo_de_grafico==2
    cla(eixos, 'reset');
    histogram(eixos,num(:,ColunaTime1(1)), 'FaceColor', cores{indice_cor1});
    hold(eixos, 'on');
    histogram(eixos,num(:,ColunaTime2(1)), 'FaceColor', cores{indice_cor2});
    hold(eixos, 'off');
    xlabel(eixos, 'Gols');
    ylabel(eixos, 'Jogos');
    title(eixos, 'Histograma');
    legend(eixos, Time1, Time2);

elseif Tipo_de_grafico==3
    cla(eixos, 'reset');
    hold(eixos, 'on');
    bar(eixos, 1, num(1,ColunaTime1(2)),cores{indice_cor1});
    bar(eixos, 2, num(1,ColunaTime2(2)), cores{indice_cor2});
    bar(eixos, 3, num(1,11), 'y');
    hold(eixos, 'off');
    xticklabels(eixos, {'', Time1, '', Time2, '', 'Empate'});
    ylabel(eixos, 'Quantidade de Jogos');
    title(eixos,'Vitórias em Clássicos');

elseif Tipo_de_grafico==4
    cla(eixos, 'reset');
    ColunaTime1=find(strcmp(texto_torcida(1,:),Time1)==1)+1;
    ColunaTime2=find(strcmp(texto_torcida(1,:),Time2)==1)+1;
    hold(eixos, 'on');
    plot(eixos,num_torcida(:,1),num_torcida(:,ColunaTime1),cores{indice_cor1});
    plot(eixos,num_torcida(:,1),num_torcida(:,ColunaTime2),cores{indice_cor2});
    hold(eixos, 'off');
    xlabel(eixos,'Ano');
    ylabel(eixos,'Publico');
    legenda2=legend(eixos, Time1, Time2);
    legenda2.Location = 'northwest';
    title(eixos, 'Historico Publico');

elseif Tipo_de_grafico==5
    cla(eixos, 'reset');
    plot(eixos,num_retrospecto(:,1),num_retrospecto(:,ColunaTime1Retro+2),cores{indice_cor1},num_retrospecto(:,1),
        num_retrospecto(:,ColunaTime2Retro+2),cores{indice_cor2});
    hold(eixos, 'on');
    ylim(eixos, [0 80]);
    ylabel(eixos, 'Pontuação');

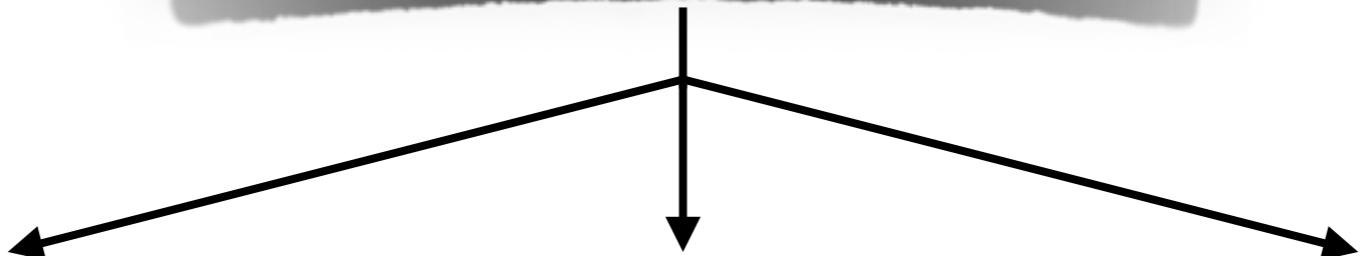
```



Fellowship of the Ring



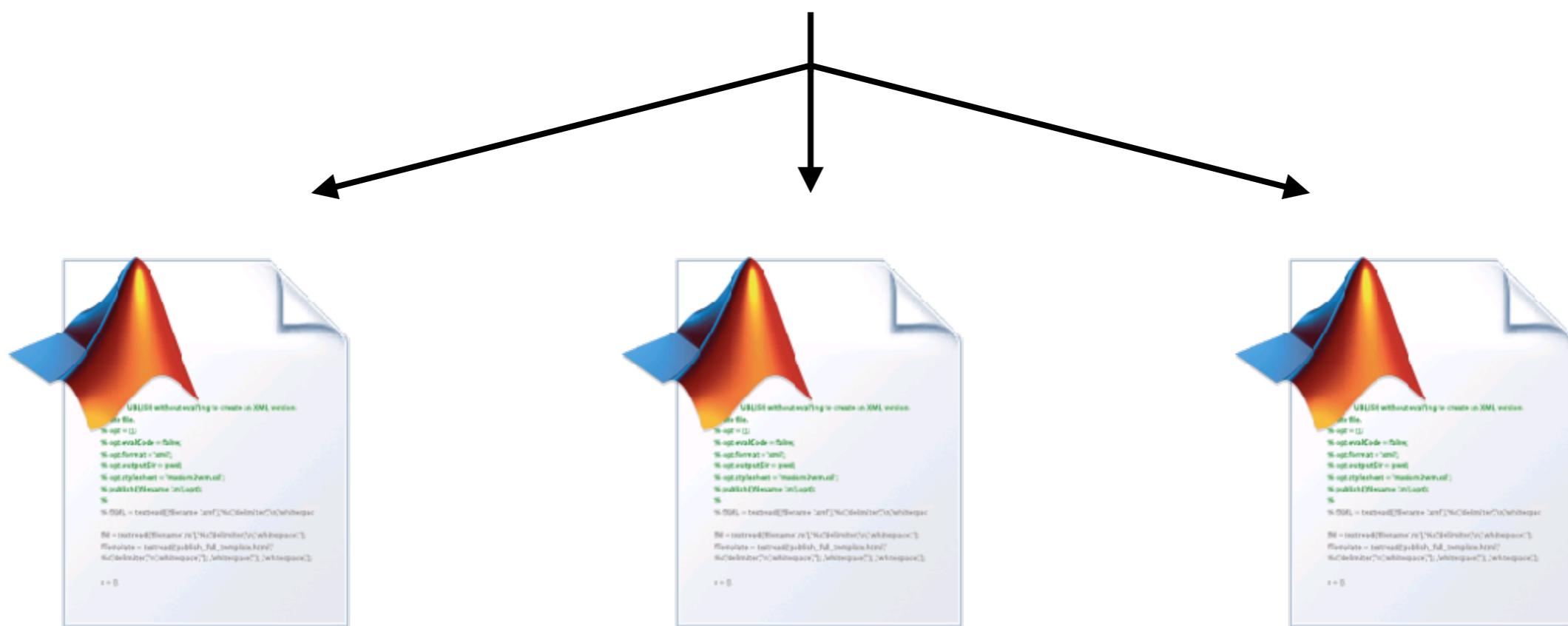
© Haley 2013 all rights reserved





```
UFLISH without waiting to create an XML version.
%file
%opt = [];
%opt.evalCode = false;
%opt.format = 'xml';
%opt.optimizeDir = '';
%opt.stylesheet = 'matlab2xml.css';
%publishFilename('nl2m');
%
%XML = buildxml('filename.xml','%Cdefm1se','c/whitepac
%
%fil = read('filename.xml','%Cdefm1se/c/whitepac');
%fildata = readxml('publish_file_template.html');
%fildata(c/c/whitepac/c/whitepac/c/whitepac);
%fildata(c/c/whitepac/c/whitepac/c/whitepac);
%
z = [];
```

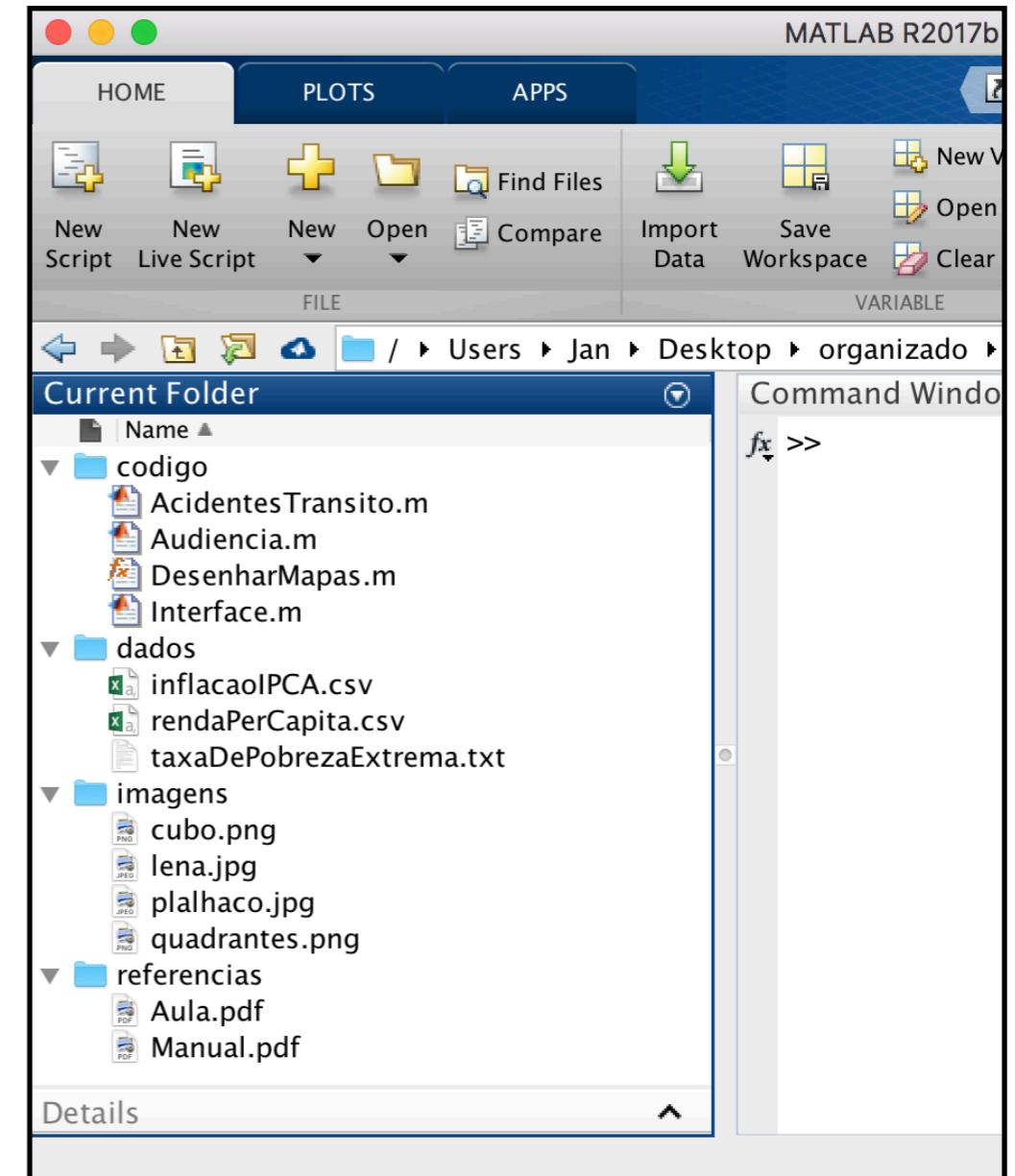
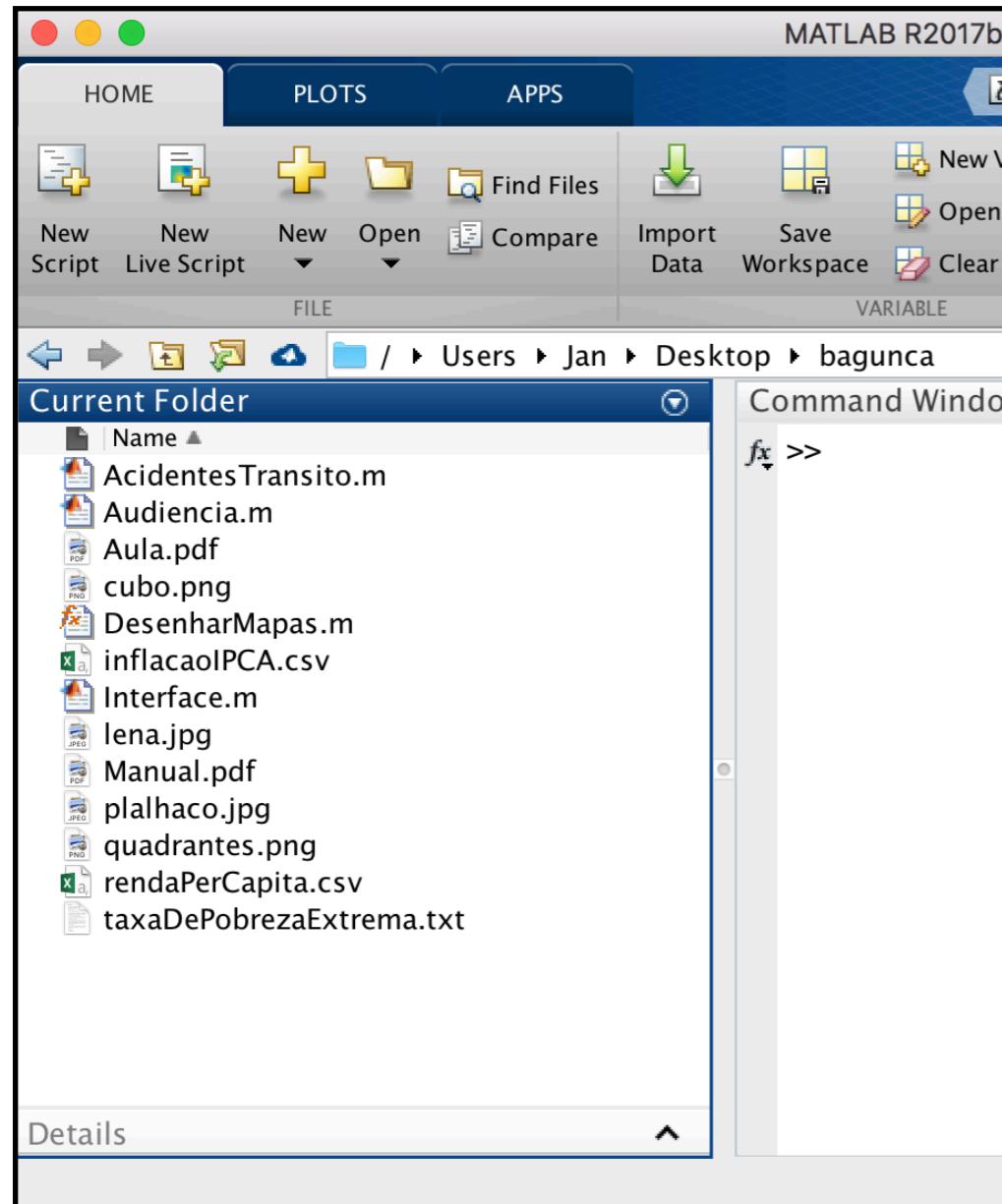
TudoJunto.m
(500 linhas)



Script1.m
(60 linhas)

Funcao1.m
(20 linhas)

Funcao2.m
(35 linhas)





**KEEP
CALM
AND
REFACTOR
CODE**