

Maroto V2

Criando PDFs de maneira simples



E ai =)

Sou Johnathan Fercher

Tecnólogo em TI

Mestre em Robótica

Desenvolvedor Backend

Contribuidor open-source

Medium + Palestras



Roadmap



1

V1



Há muito tempo atrás... existia o TMS



Remito de Outbound

XPR1

 **mercado
livre**


223651872569267

MERCADO ENVIOS

Data de início: 23/03/2022 10:43:31	Doca: 4009
Data de finalização: 23/03/2022 11:38:59	Placa: <input type="text"/>
Zonas: spr3	Pacotes: 2223
Lacre: 328275	Responsável: <input type="text"/>

Insumos utilizados:
gaylord: 23
manga pallet: 0
pallet: 25

Zona	Número de HU	Pacotes
spr3	2236447457564659	220
spr3	2236459425427095	118
spr3	2236444287557941	137
spr3	2236443799956880	180
spr3	223645275539994	8
spr3	2236454615737378	17
spr3	2236326547205545	25
spr3	2236462979667228	140
spr3	2236460933946156	4
spr3	2236462047252758	6
spr3	2236447835526945	210
spr3	2236438032608629	139
spr3	2236433640915271	5
spr3	2236454331344827	185
spr3	2236434900252916	21
spr3	2236433282964747	130
spr3	2236466991378815	1
spr3	223645827989996	127
spr3	2236438445383245	165
spr3	2236449349878939	200

Gofpdf

```
if carrierLogo != "" {
    pdf.Image(imageAppBaseDir+carrierLogo, x: 12, y: 34, w: 38, h: 13, flow: false, tp: "", link: 0, linkStr: "")
} else {
    left, _, right, _ := pdf.GetMargins()
    pdfSetFont( familyStr: "Arial", styleStr: "B", size: 12)
    pdf.SetLeftMargin( margin: 20)
    pdf.SetRightMargin( margin: 140)
    pdf.CellFormat( w: 0, h: 60, carrierName, borderStr: "", ln: 0, alignStr: "", fill: false, link: 0, linkStr: "")
    pdfSetFont( familyStr: "Arial", styleStr: "", size: 8)
    pdf.SetLeftMargin(left)
    pdf.SetRightMargin(right)
}
```

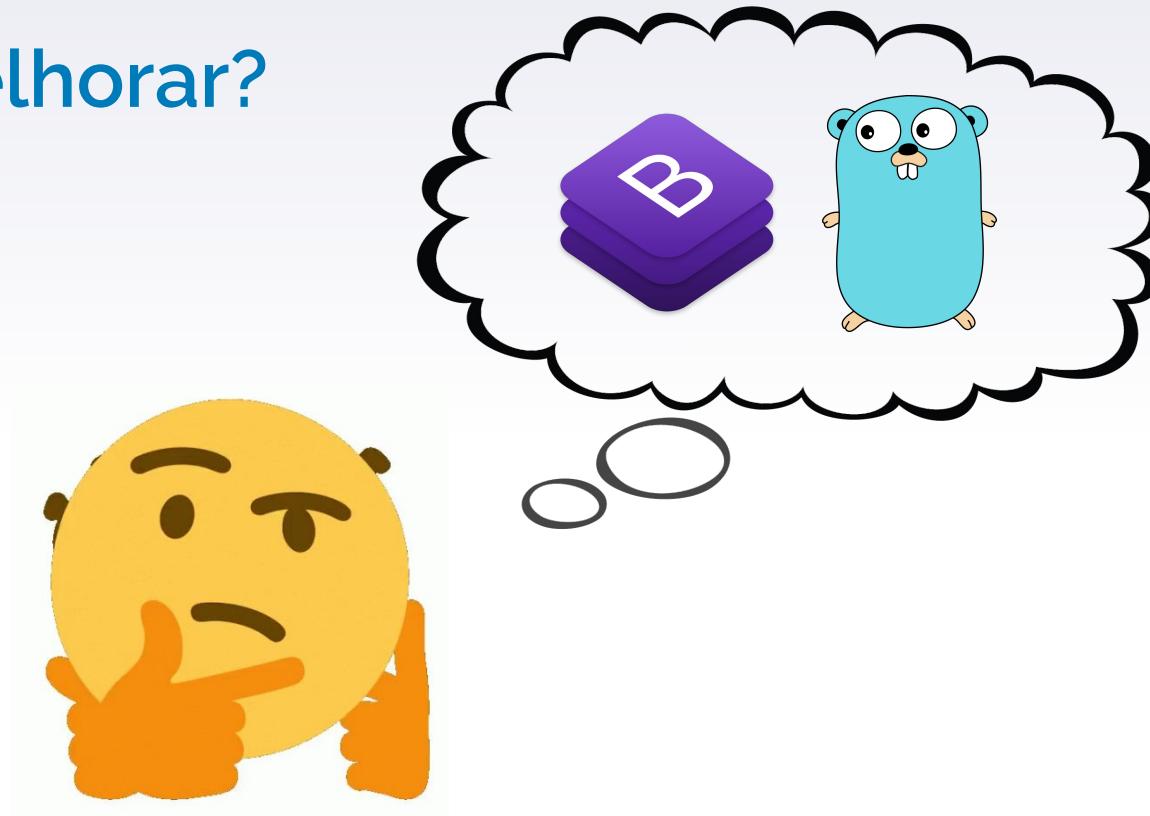


PDFs naquela época

Lib	Estrelas	Contribuidores	Licença
jung-kurt/gofpdf	3.324	56	MIT
unidoc/unipdf	513	22	\$\$\$
SebastiaanKlippert/go-wkhtmltopdf	330	6	MIT

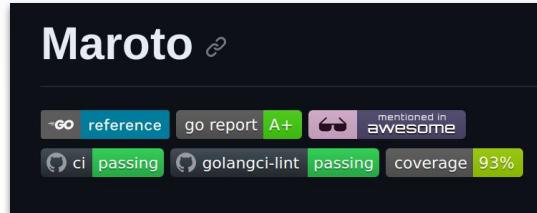


Como melhorar?



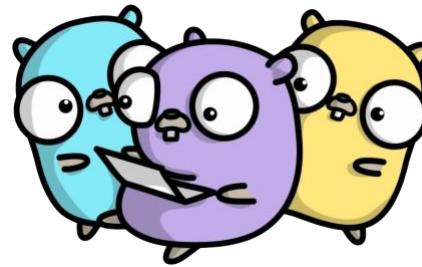
Fundamentos do Maroto

1. Sistema de Grid;
2. Simples, Leve e Rápido;
3. Escopo pequeno (Foco principal MELI);
4. Qualidade e Documentação;
5. Licença MIT;



Primeiro esboço

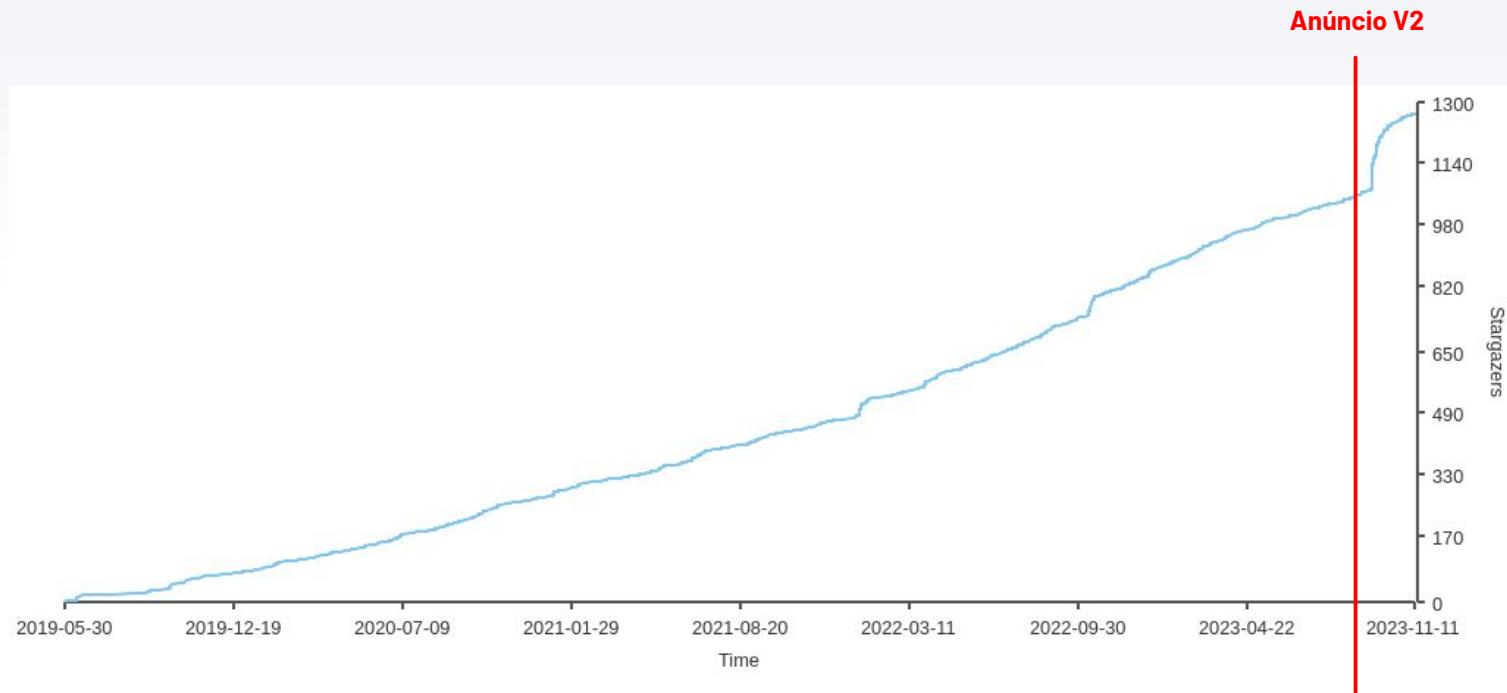
```
m.Row( height: 20, func() {
    m.Col( width: 6, func() {
        _ = m.Base64Image(meliLogoResource.Base64, meliLogoResource.Extension, props.Rect{
            Percent: 90,
            Center: true,
        })
    })
    m.Col( width: 6, func() {
        m.Text(request.Id, props.Text{
            Top: 15,
            Size: 9,
            Align: consts.Center,
        })
    })
})
```



Timeline V1



Crescimento



Problemas V1

The screenshot shows the GitHub Issues page for the repository `johnfercher / maroto`. The navigation bar includes links for Code, Issues (16), Pull requests (1), Discussions, Actions, Projects (1), Wiki, and more. The search bar contains the query `is:issue is:closed sort:comments-desc`. Below the search bar are buttons for Filters, Labels (15), Milestones (0), and New issue. A link to Clear current search query, filters, and sorts is present. The main area displays a list of 96 closed issues, with the first few listed below:

- Add custom page size option when creating documents** (good first issue, help wanted, new feature) - #148 by sanderaldo was closed on May 22, 2020. 1 comment, 8 reactions.
- m.Text 自动换行, 对中文没有支持** - #212 by xijaja was closed 3 weeks ago. 7 comments.
- Why a dot is printed instead of ... ?** (question, wontfix) - #205 by frederikhors was closed on May 8, 2022. 6 comments.
- A way to truncate text?** (help wanted, new feature) - #204 by frederikhors was closed 2 weeks ago. 5 comments.
- It does not support Chinese and Arabic characters.** - #142 by gonzalezljesus was closed on Feb 4, 2021. 1 comment, 5 reactions.
- Migrate to Go 1.13 in travis tests** (good first issue, hacktoberfest, help wanted, technical debt) - #108 by SuddenGunter was closed on Nov 11, 2019. 5 comments.

Problemas V1

m.Text 自动换行，对中文没有支持 #212

Closed xijaja opened this issue on May 26, 2022 · 7 comments

xijaja commented on May 26, 2022

我尝试使用下面的代码，但其实，只有英文自动换行：

```
m.Row(130, func() {
    m.Col(12, func() {
        text1 := "此工廠診斷報告僅供給工廠作改善依據，"
        text2 := "This factory audit report is only intended for the factory to improve, "
        m.Text(text1+text2, props.Text{
            Size: 12,
            Top: 10,
            Align: consts.Left,
            // Extrapolate: false,
            VerticalPadding: 2.0,
        })
    })
})
```

不论我是否在 text1 的字符串中添加“\n”，结果都是只有英文自动换行；并且，如果我仅使用 text1 结果也是一样，它并不会自动换行，您可以尝试一下。

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestone: No milestone

Development: Create a branch for this issue or link a pull request.

Notifications: Customize

Unsubscribe

You're receiving notifications because you're watching this repository.

Problemas V1

```
// Maroto is the principal abstraction to create a PDF document.
type Maroto interface {
    // Grid System
    Row(height float64, closure func())
    Col(width uint, closure func())
    ColSpace(gridSize uint)

    // Registers
    RegisterHeader(closure func())
    RegisterFooter(closure func())

    // Outside Col/Row Components
    TableList(header []string, contents [][]string, prop ...props.TableList)
    Line(spaceHeight float64, prop ...props.Line)

    // Inside Col/Row Components
    Text(text string, prop ...props.Text)
    FileImage(filePathName string, prop ...props.Rect) (err error)
    Base64Image(base64 string, extension consts.Extension, prop ...props.Rect) (err error)
    Barcode(code string, prop ...props.Barcode) error
    QrCode(code string, prop ...props.Rect)
    DataMatrixCode(code string, prop ...props.Rect)
    Signature(label string, prop ...props.Font)

    // File System
    OutputFileAndClose(filePathName string) error
    Output() (bytes.Buffer, error)
}
```

```
// Helpers
AddPage()
SetBorder(on bool)
SetBackgroundColor(color color.Color)
SetAliasNbPages(alias string)
SetFirstPageNb(number int)
GetBorder() bool
GetPageSize() (width float64, height float64)
GetCurrentPage() int
GetCurrentOffset() float64
SetPageMargins(left, top, right float64)
GetPageMargins() (left float64, top float64, right float64, bottom float64)
SetMaxGridSum(maxGridSum float64)

// Fonts
AddUTF8Font(familyStr string, styleStr consts.Style, fileStr string)
AddUTF8FontFromBytes(familyStr string, styleStr consts.Style, utf8Bytes []byte)
SetFontLocation(fontDirStr string)
SetDefaultFontFamily(fontFamily string)
GetDefaultFontFamily() string

// Metadata
SetCompression(compress bool)
SetProtection(actionFlag byte, userPassStr, ownerPassStr string)
SetAuthor(author string, isUTF8 bool)
SetCreator(creator string, isUTF8 bool)
SetSubject(subject string, isUTF8 bool)
SetTitle(title string, isUTF8 bool)
SetCreationDate(time time.Time)
}
```

Problemas V1

```
// Row define a row and enable add columns inside the row.  
// Maroto do not support recursive rows or rows inside columns.  
func (s *PdfMaroto) Row(height float64, closure func()) {  
    // Used to calculate the height of the footer.  
    if s.calculationMode {  
        s.footerHeight += height  
        return  
    }  
  
    _, pageHeight := s.Pdf.GetPageSize()  
    _, top, _, bottom := s.Pdf.GetMargins()  
  
    totalOffsetY := int(s.offsetY + height + s.footerHeight)  
    maxOffsetPage := int(pageHeight - bottom - top)  
  
    // Note: The headerFooterContextActive is needed to avoid recursive  
    // calls without end, because footerClosure and headerClosure actually  
    // have Row calls too.  
  
    // If the new cell to be added pass the useful space counting the  
    // height of the footer, add the footer.  
    if totalOffsetY > maxOffsetPage {  
        if !s.headerFooterContextActive {  
            s.headerFooterContextActive = true  
            s.footer()  
            s.headerFooterContextActive = false  
            s.offsetY = 0  
            s.pageIndex++  
        }  
    }  
  
    // If is a new page, add the header.  
    if !s.headerFooterContextActive {  
        if s.offsetY == 0 {  
            s.headerFooterContextActive = true  
            s.header()  
            s.headerFooterContextActive = false  
        }  
    }  
  
    s.rowHeight = height  
    s.xColOffset = 0  
  
    // This closure has the Cols to be executed.  
    closure()  
  
    s.offsetY += s.rowHeight  
    s.Pdf.Ln(s.rowHeight)  
}
```

Problemas V1

This repository has been archived by the owner on Nov 13, 2021. It is now read-only.

A PDF document generator with high level support for text, drawing and images

godoc.org/github.com/jung-kurt/gofpdf

MIT license

4.2k stars 762 forks 107 watching Activity

Public archive repository



Problemas V1

1. Altamente dependente do Gofpdf;
2. Sem suporte a testes unitários;
3. Sem suporte a geração paralela;
4. Mais suscetível a erro de runtime;
5. Dificuldade de adicionar novas features;

Resultados V1

1. 1163 estrelas;
2. 152 forks;
3. 22 contribuidores;
4. 73 issues resolvidas, 39 abertas;
5. 51 releases;
6. 136 pull requests;
7. 265 usos no github;

Resultados V1

1. 4 milhões de PDFs gerados em FOS/TMS (**Dados até Março de 2022**);



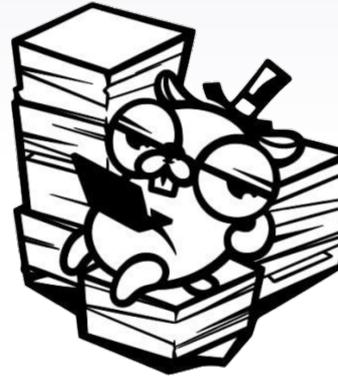
[8962 stars](#)



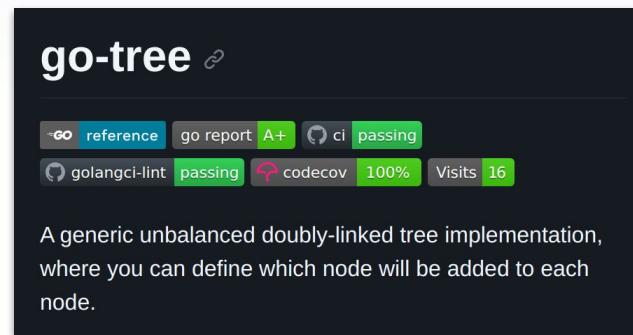
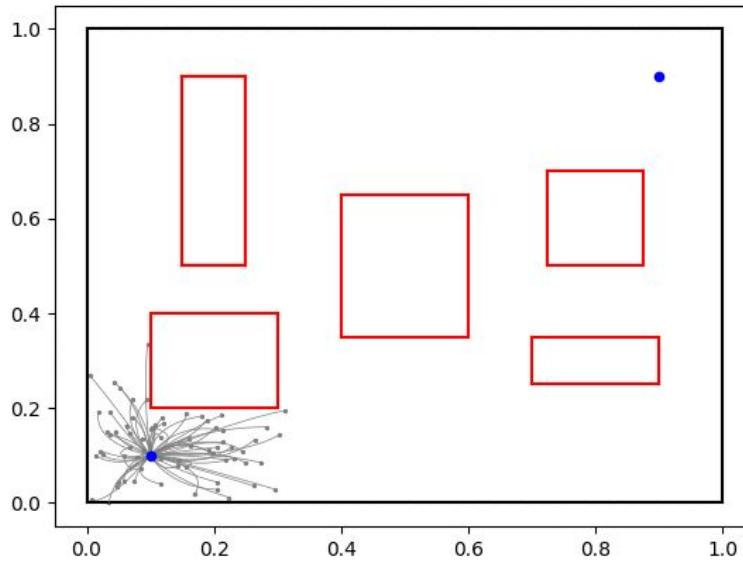
[1719 stars](#)

2

V2



Um final de semana qualquer...



Source: <https://www.cs.cmu.edu/~motionplanning/lecture/lec20.pdf>

Final de Domingo...



Final de Domingo...

[RFC] Maroto V2

Summary: Improve usability, allow unit testing, allow parallelism, allow generation based on HTML/JSON, allow generation of other documents as ZPL.

Created: Sep 11, 2023

Status: WIP | In-Review | **Approved** | Obsolete

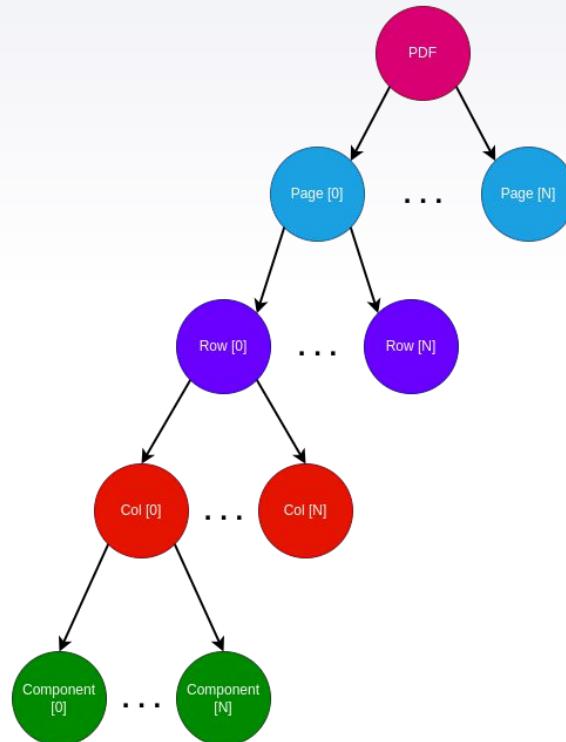
Current Version: 0.42.0

Owner: Johnathan Fercher

Target Version: 2.0

Contributors: Artem Kolomytsev, Felipe Amaral + anyone
who wants to join the team.

Árvore de Componentes



Novo Core

```
type Provider interface {
    // Grid
    CreateRow(height float64)
    CreateCol(width, height float64, config *entity.Config, prop *props.Cell)

    // Features
    AddLine(cell *entity.Cell, prop *props.Line)
    AddText(text string, cell *entity.Cell, prop *props.Text)
    GetTextHeight(prop *props.Font) float64
    AddMatrixCode(code string, cell *entity.Cell, prop *props.Rect)
    AddQrCode(code string, cell *entity.Cell, rect *props.Rect)
    AddBarcode(code string, cell *entity.Cell, prop *props.Barcode)
    AddImageFromFile(value string, cell *entity.Cell, prop *props.Rect)
    AddImageFromBytes(bytes []byte, cell *entity.Cell, prop *props.Rect, extension extension.Type)
    AddBackgroundImageFromBytes(bytes []byte, cell *entity.Cell, prop *props.Rect, extension extension.Type)

    // General
    GenerateBytes() ([]byte, error)

    SetProtection(protection *entity.Protection)
    SetCompression(compression bool)
    SetMetadata(metadata *entity.Metadata)
}
```

Novo Core

```
type Maroto interface {
    RegisterHeader(rows ...Row) error
    RegisterFooter(rows ...Row) error
    AddRows(rows ...Row)
    AddRow(rowHeight float64, cols ...Col) Row
    AddPages(pages ...Page)
    GetStructure() *node.Node[Structure]
    Generate() (Document, error)
}

type Document interface {
    GetBytes() []byte
    GetBase64() string
    Save(file string) error
    GetReport() *metrics.Report
    Merge([]byte) error
}

type Node interface {
    SetConfig(config *entity.Config)
    GetStructure() *node.Node[Structure]
}

type Component interface {
    Node
    Render(provider Provider, cell *entity.Cell)
}
```



```
type Col interface {
    Node
    Add(components ...Component) Col
    GetSize() int
    WithStyle(style *props.Cell) Col
    Render(provider Provider, cell entity.Cell, createCell bool)
}

type Row interface {
    Node
    Add(cols ...Col) Row
    GetHeight() float64
    WithStyle(style *props.Cell) Row
    Render(provider Provider, cell entity.Cell)
}

type Page interface {
    Node
    Add(rows ...Row) Page
    GetRows() []Row
    GetNumber() int
    SetNumber(number int, total int)
    Render(provider Provider, cell entity.Cell)
}
```

Novo Core

```
type Maroto interface {
    RegisterHeader(rows ...Row) error
    RegisterFooter(rows ...Row) error
    AddRows(rows ...Row)
    AddRow(rowHeight float64, cols ...Col) Row
    AddPages(pages ...Page)
    GetStructure() *node.Node[Structure]
    Generate() (Document, error)
}

type Document interface {
    GetBytes() []byte
    GetBase64() string
    Save(file string) error
    GetReport() *metrics.Report
    Merge([]byte) error
}

type Node interface {
    SetConfig(config *entity.Config)
    GetStructure() *node.Node[Structure]
}

type Component interface {
    Node
    Render(provider Provider, cell *entity.Cell)
}
```



```
type Col interface {
    Node
    Add(components ...Component) Col
    GetSize() int
    WithStyle(style *props.Cell) Col
    Render(provider Provider, cell entity.Cell, createCell bool)
}

type Row interface {
    Node
    Add(cols ...Col) Row
    GetHeight() float64
    WithStyle(style *props.Cell) Row
    Render(provider Provider, cell entity.Cell)
}

type Page interface {
    Node
    Add(rows ...Row) Page
    GetRows() []Row
    GetNumber() int
    SetNumber(number int, total int)
    Render(provider Provider, cell entity.Cell)
}
```

Novo Core

```
type Maroto interface {
    RegisterHeader(rows ...Row) error
    RegisterFooter(rows ...Row) error
    AddRows(rows ...Row)
    AddRow(rowHeight float64, cols ...Col) Row
    AddPages(pages ...Page)
    GetStructure() *node.Node[Structure]
    Generate() (Document, error)
}

type Document interface {
    GetBytes() []byte
    GetBase64() string
    Save(file string) error
    GetReport() *metrics.Report
    Merge([]byte) error
}

type Node interface {
    SetConfig(config *entity.Config)
    GetStructure() *node.Node[Structure]
}

type Component interface {
    Node
    Render(provider Provider, cell *entity.Cell)
}
```



```
type Col interface {
    Node
    Add(components ...Component) Col
    GetSize() int
    WithStyle(style *props.Cell) Col
    Render(provider Provider, cell entity.Cell, createCell bool)
}

type Row interface {
    Node
    Add(cols ...Col) Row
    GetHeight() float64
    WithStyle(style *props.Cell) Row
    Render(provider Provider, cell entity.Cell)
}

type Page interface {
    Node
    Add(rows ...Row) Page
    GetRows() []Row
    GetNumber() int
    SetNumber(number int, total int)
    Render(provider Provider, cell entity.Cell)
}
```



Novo Core



```
type Maroto interface {
    RegisterHeader(rows ...Row) error
    RegisterFooter(rows ...Row) error
    AddRows(rows ...Row)
    AddRow(rowHeight float64, cols ...Col) Row
    AddPages(pages ...Page)
    GetStructure() *node.Node[Structure]
    Generate() (Document, error)
}

type Document interface {
    GetBytes() []byte
    GetBase64() string
    Save(file string) error
    GetReport() *metrics.Report
    Merge([]byte) error
}

type Node interface {
    SetConfig(config *entity.Config)
    GetStructure() *node.Node[Structure]
}

type Component interface {
    Node
    Render(provider Provider, cell *entity.Cell)
}
```

```
type Col interface {
    Node
    Add(components ...Component) Col
    GetSize() int
    WithStyle(style *props.Cell) Col
    Render(provider Provider, cell entity.Cell, createCell bool)
}

type Row interface {
    Node
    Add(cols ...Col) Row
    GetHeight() float64
    WithStyle(style *props.Cell) Row
    Render(provider Provider, cell entity.Cell)
}

type Page interface {
    Node
    Add(rows ...Row) Page
    GetRows() []Row
    GetNumber() int
    SetNumber(number int, total int)
    Render(provider Provider, cell entity.Cell)
}
```

Componentes

Name	Last commit message
..	
code	Re-organize test files (#338)
col	Add tests on maroto level (#342)
image	Re-organize packages (#351)
line	Re-organize test files (#338)
list	Add tests on maroto level (#342)

```
func NewQr(code string, barcodeProps ...props.Rect) core.Component {
    prop := props.Rect{}
    if len(barcodeProps) > 0 {
        prop = barcodeProps[0]
    }
    prop.MakeValid()

    return &qrCode{
        code: code,
        prop: prop,
    }
}

func NewQrCol(size int, code string, ps ...props.Rect) core.Col {
    qrCode := NewQr(code, ps...)
    return col.New(size).Add(qrCode)
}

func NewQrRow(height float64, code string, ps ...props.Rect) core.Row {
    qrCode := NewQr(code, ps...)
    c := col.New().Add(qrCode)
    return row.New(height).Add(c)
}
```

Exemplo

```
func main() {
    m := maroto.New()

    m.AddRow(20,
        code.NewBarCol(4, "barcode"),
        code.NewMatrixCol(4, "matrixcode"),
        code.NewQrCol(4, "qrcode"),
    )

    m.AddRow(10, col.New(12))

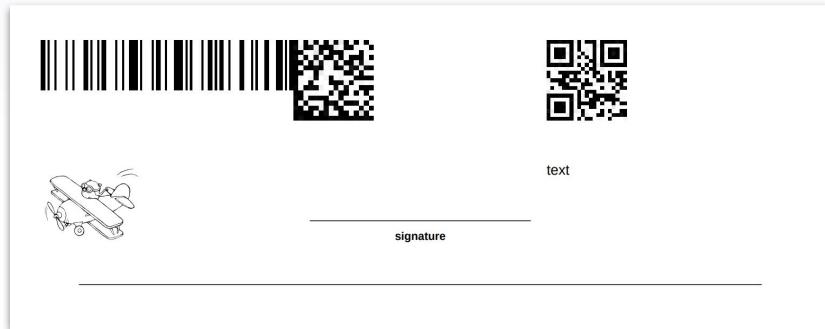
    m.AddRow(20,
        image.NewFromFileCol(4, "docs/assets/images/biplane.jpg"),
        signature.NewCol(4, "signature"),
        text.NewCol(4, "text"),
    )

    m.AddRow(10, col.New(12))

    m.AddRow(20, line.NewCol(12))

    document, err := m.Generate()
    if err != nil {
        log.Fatal(err)
    }

    err = document.Save("docs/assets/pdf/simpletv2.pdf")
    if err != nil {
        log.Fatal(err)
    }
}
```



Builder para Customização Global

```
type Builder interface {
    WithPageSize(size pagesize.Type) Builder
    WithDimensions(width float64, height float64) Builder
    WithMargins(left float64, top float64, right float64) Builder
    WithWorkerPoolSize(poolSize int) Builder
    WithDebug(on bool) Builder
    WithMaxGridSize(maxGridSize int) Builder
    WithDefaultFont(font *props.Font) Builder
    WithPageNumber(pattern string, place props.Place) Builder
    WithProtection(protectionType protection.Type, userPassword, ownerPassword string) Builder
    WithCompression(compression bool) Builder
    WithOrientation(orientation orientation.Type) Builder
    WithAuthor(author string, isUTF8 bool) Builder
    WithCreator(creator string, isUTF8 bool) Builder
    WithSubject(subject string, isUTF8 bool) Builder
    WithTitle(title string, isUTF8 bool) Builder
    WithCreationDate(time time.Time) Builder
    WithCustomFonts([]*entity.CustomFont) Builder
    WithBackgroundImage([]byte, extension.Type) Builder
    Build() *entity.Config
}
```

Decorator de Métricas

```
func main() {
    cfg := config.NewBuilder().
        WithCompression(true).
        Build()

    mrt := maroto.New(cfg)
    m := maroto.NewMetricsDecorator(mrt)
```

```
generate -> avg: 19.66ms, executions: [19.66ms]
header -> avg: 228.00ns, executions: [228.00ns]
footer -> avg: 56.00ns, executions: [56.00ns]
add_row -> avg: 212.50ns, executions: [107.00ns, 318.00ns]
add_cols -> avg: 184.00ns, executions: [146.00ns, 222.00ns]
file_size -> 223.68Kb
```

TableList Generic Builder

```
type Listable interface {
    GetHeader() core.Row
    GetContent(i int) core.Row
}

func BuildFromPointer[T Listable](arr []*T) ([]core.Row, error) {
}...

func Build[T Listable](arr []T) ([]core.Row, error) {
}...
```

```
func (o Object) GetHeader() core.Row {
    return row.New(10).Add(
        text.NewCol(4, "Key", props.Text{Style: fontstyle.Bold}),
        text.NewCol(8, "Bytes", props.Text{Style: fontstyle.Bold}),
    )
}

func (o Object) GetContent(i int) core.Row {
    r := row.New(5).Add(
        text.NewCol(4, o.Key),
        text.NewCol(8, o.Value),
    )

    if i%2 == 0 {
        r.WithStyle(&props.Cell{
            BackgroundColor: background,
        })
    }

    return r
}
```

Testes Unitários

maroto / .maroto.yml 

 johnfercher Refactor test package to work with files

1 lines (1 loc) · 25 Bytes  Code 55% faster with GitHub

[Code](#) [Blame](#)

```
1     test_path: "test/maroto/"
```

```
func TestMaroto_GetStructure(t *testing.T) {
    // Arrange
    m := maroto.New()

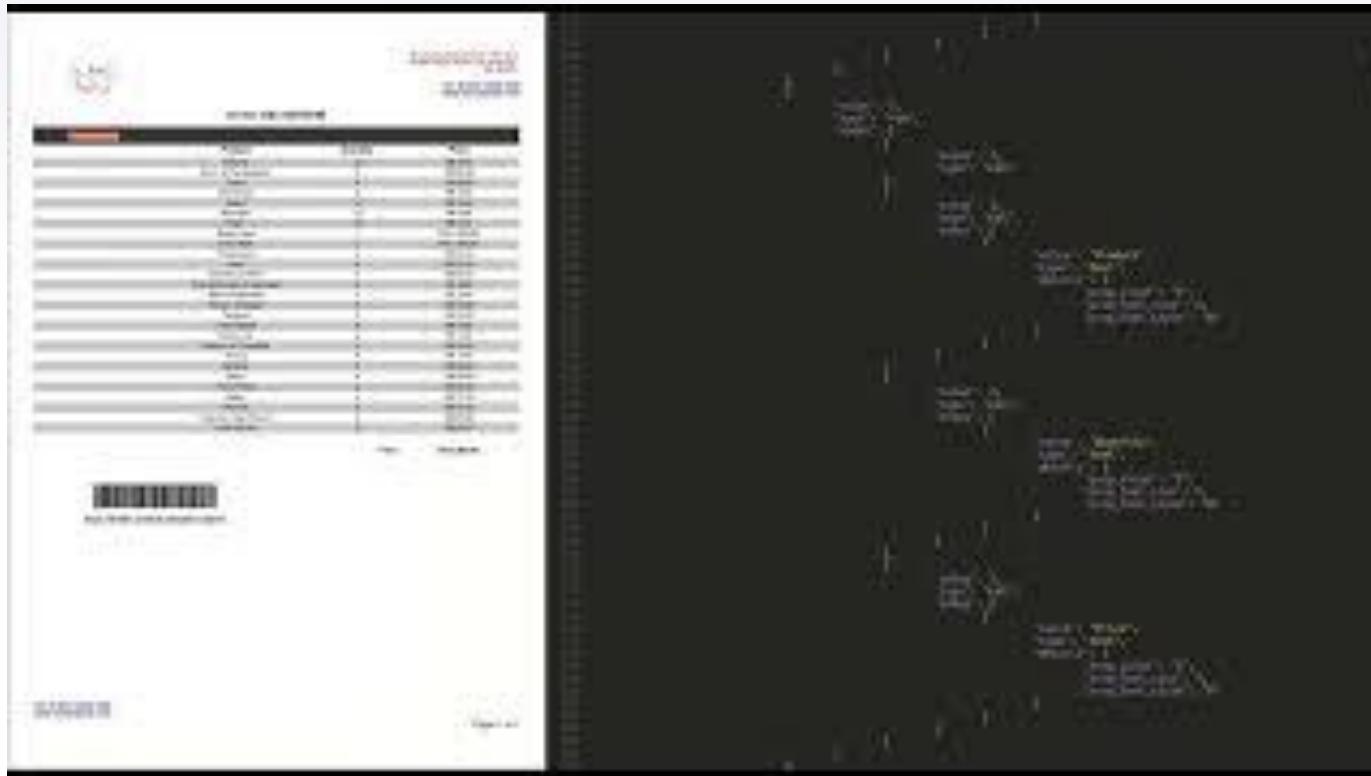
    m.AddRow(10,
        code.NewBarCol(4, "barcode"),
        code.NewMatrixCol(4, "matrixcode"),
        code.NewQrCol(4, "qrcode"),
    )

    m.AddRow(10,
        image.NewFromFileCol(3, "image"),
        image.NewFromBytesCol(3, []byte{0, 1, 2}, extension.Png),
        signature.NewCol(3, "signature"),
        text.NewCol(3, "text"),
    )

    m.AddRow(10, line.NewCol(12))

    // Assert
    test.New(t).Assert(m.GetStructure()).Equals("example_unit_test.json")
}
```

Testes Unitários



3

$V_1 \times V_2$



Código

maroto / internal / examples / barcodegrid / main.go

```
Code Blame 104 lines (95 loc) · 2.05 KB ⚡ Code 55% faster with GitHub Copilot

13 func main() {
81     })
82     m.col(4, func() {
83         m.Barcode("https://github.com/johnfercher/maroto", props.Barcode{
84             Center: true,
85             Percent: 75,
86         })
87     })
88     m.col(2, func() {
89         m.Barcode("https://github.com/johnfercher/maroto", props.Barcode{
90             Center: true,
91             Percent: 100,
92         })
93     })
94 }
95
96 err := m.OutputFileAndClose("internal/examples/pdfs/barcodegrid.pdf")
97 if err != nil {
98     fmt.Println("Could not save PDF:", err)
99     os.Exit(1)
100 }
101
102 end := time.Now()
103 fmt.Println(end.Sub(begin))
104 }
```

maroto / docs / assets / examples / barcodegrid / v2 / main.go

```
Code Blame 90 lines (77 loc) · 1.97 KB ⚡ Code 55% faster with GitHub Copilot

14 func main() {
67     Center: true,
68     Percent: 75,
69 },
70 code.NewBarCol(2, "https://github.com/johnfercher/maroto", props.Barcode{
71     Center: true,
72     Percent: 100,
73 }),
74 )
75
76 document, err := m.Generate()
77 if err != nil {
78     log.Fatal(err.Error())
79 }
80
81 err = document.Save("docs/assets/pdf/barcodegridv2.pdf")
82 if err != nil {
83     log.Fatal(err.Error())
84 }
85
86 err = document.GetReport().Save("docs/assets/text/barcodegridv2.txt")
87 if err != nil {
88     log.Fatal(err.Error())
89 }
90 }
```

Performance

1 - Barcode Grid

- V1 - 3.83ms
- V2 - 1.13ms

2 - Billing Example (100 páginas)

- V1 - 7.78s
- V2 - 2.75s

Processador: I9 13900kf

Memória: 32GB DDR5 5.6GHz

Performance

1 - Barcode Grid

- V1 - 3.83ms
- V2 - 1.13ms

2 - Billing Example (100 páginas)

- V1 - 7.78s
- V2 - 2.75s
- V2-Parallel - 546ms (**7 go-routines**)

Processador: I9 13900kf

Memória: 32GB DDR5 5.6GHz

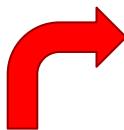
Performance

1 - Barcode Grid

- V1 - 3.83ms
- V2 - 1.13ms

2 - Billing Example (100 páginas)

- V1 - 7.78s
- V2 - 2.75s
- V2-Parallel - 546ms (**7 go-routines**)



  coverage 100%

go-async

Repository that implements goroutines abstractions, making easy to run async code in Go.

Processador: I9 13900kf

Memória: 32GB DDR5 5.6GHz

Estado de Desenvolvimento V2

1. Após 42 versões alpha
2. V2.0.0-beta.3
3. Grande diminuição na quantidade de issues em aberto
4. Nova documentação: <https://maroto.io>
5. Piloto do V2: <https://github.com/kubescape/kubescape/issues/1458>

Obrigado!

Perguntas?

Você pode encontrar essa apresentação
em:

- ▶ <https://johnfercher.com/#/pt-br/talks>

