

GUIA COMPLETO: LIVRO INTERATIVO DOS MANUAIS FABLAB EM R






VISÃO GERAL

Este guia mostra como criar um livro digital interativo com os manuais das máquinas do FABLAB usando R Studio, com três abordagens principais:

1. **bookdown** - Livro técnico profissional
2. **flexdashboard** - Dashboard interativo
3. **shiny** - Aplicação web completa

ABORDAGEM 1: BOOKDOWN (RECOMENDADA)

Por que bookdown?

-  Ideal para documentação técnica
-  Navegação por capítulos
-  Busca integrada
-  Exportação para PDF, HTML, EPUB
-  Referências cruzadas automáticas

1.1 Instalação e Setup

Plain Text

```
# Instalar pacotes necessários
install.packages(c("bookdown", "rmarkdown", "knitr", "DT", "plotly",
"htmlwidgets"))

# Carregar bibliotecas
library(bookdown)
library(rmarkdown)
library(knitr)
library(DT)
library(plotly)
```

1.2 Estrutura do Projeto

Plain Text

```
fablab_manual/  
├─ index.Rmd          # Página inicial  
├─ 01-introducao.Rmd  # Introdução geral  
├─ 02-impressoras-3d.Rmd # Manuais impressoras 3D  
├─ 03-laser.Rmd       # Manual cortadora laser  
├─ 04-router-cnc.Rmd  # Manual Router CNC  
├─ 05-plotter.Rmd     # Manual Plotter  
├─ 06-pcb.Rmd         # Manuais fabricação PCB  
├─ 07-manutencao.Rmd  # Cronogramas de manutenção  
├─ 08-troubleshooting.Rmd # Solução de problemas  
├─ _bookdown.yml      # Configuração do livro  
├─ _output.yml        # Configuração de saída  
├─ book.bib           # Bibliografia  
├─ style.css          # Estilos personalizados  
└─ images/            # Imagens e diagramas  
    ├─ logos/  
    ├─ maquinas/  
    └─ diagramas/
```

1.3 Arquivo de Configuração (_bookdown.yml)

YAML

```
book_filename: "manual_fablab_ufpb"  
language:  
  ui:  
    chapter_name: "Capítulo "  
  
delete_merged_file: true  
  
rmd_files: [  
  "index.Rmd",  
  "01-introducao.Rmd",  
  "02-impressoras-3d.Rmd",  
  "03-laser.Rmd",  
  "04-router-cnc.Rmd",  
  "05-plotter.Rmd",  
  "06-pcb.Rmd",  
  "07-manutencao.Rmd",  
  "08-troubleshooting.Rmd"  
]
```

1.4 Configuração de Saída (_output.yml)

YAML

```
bookdown::gitbook:
  css: style.css
  config:
    toc:
      before: |
        <li><a href=".">Manual FABLAB UFPB</a></li>
      after: |
        <li><a href="https://github.com/rstudio/bookdown"
target="blank">Powered by bookdown</a></li>
  edit: https://github.com/fablab-ufpb/manual/edit/master/%s
  download: ["pdf", "epub"]
  search: yes
  sharing:
    facebook: yes
    github: yes
    twitter: yes
    linkedin: yes
    weibo: no
    instapaper: no
    vk: no
    all: ['facebook', 'github', 'twitter', 'linkedin']

bookdown::pdf_book:
  includes:
    in_header: preamble.tex
  latex_engine: xelatex
  citation_package: natbib
  keep_tex: yes

bookdown::epub_book: default
```

1.5 Página Principal (index.Rmd)

Markdown

```
---
title: "Manual Interativo das Máquinas"
subtitle: "FABLAB - Universidade Federal da Paraíba"
author: "Equipe FABLAB UFPB"
date: "`r Sys.Date()`"
site: bookdown::bookdown_site
documentclass: book
bibliography: [book.bib]
biblio-style: apalike
link-citations: yes
```

```
description: "Manual completo e interativo das máquinas do FABLAB UFPB"
```

```
---
```

```
# Bem-vindos ao FABLAB UFPB {-}
```

```
```{r setup, include=FALSE}
```

```
knitr::opts_chunk$set(echo = FALSE, message = FALSE, warning = FALSE)
```

```
library(DT)
```

```
library(plotly)
```

```
library(knitr)
```

<div style="text-align: center;">



</div>

## Sobre este Manual {-}

Este manual interativo contém todas as informações necessárias para operar, manter e solucionar problemas das máquinas do FABLAB UFPB.

## Características do Manual {-}

- 📱 **Responsivo:** Funciona em desktop, tablet e mobile
- 🔍 **Busca Integrada:** Encontre informações rapidamente
- 📊 **Conteúdo Interativo:** Gráficos, tabelas e simulações
- 📄 **Download:** Disponível em PDF e EPUB
- 🔗 **Referências Cruzadas:** Navegação inteligente entre seções

## Como Usar {-}

1. **Navegação:** Use o menu lateral para acessar capítulos
2. **Busca:** Digite termos na caixa de busca no topo
3. **Interatividade:** Clique em gráficos e tabelas para explorar
4. **Download:** Use os botões no topo para baixar versões offline

## Equipamentos do FABLAB {-}

Plain Text

```
equipamentos <- data.frame(
 Equipamento = c("STORM 1390", "Prusa i3 MK2", "MakerBot Replicator 2X",
 "XYZprinting Nobel 1.0A", "Ender 5 Plus", "Router CNC",
 "Plotter VISUTECH", "Bungard CCD/2/ATC"),
 Categoria = c("Corte a Laser", "Impressão 3D FDM", "Impressão 3D FDM",
 "Impressão 3D SLA", "Impressão 3D FDM", "Usinagem CNC",
 "Corte de Vinil", "Fabricação PCB"),
 Status = c("Operacional", "Operacional", "Manutenção",
 "Operacional", "Operacional", "Operacional",
 "Operacional", "Operacional"),
 Capitulo = c("Capítulo 3", "Capítulo 2", "Capítulo 2",
 "Capítulo 2", "Capítulo 2", "Capítulo 4",
 "Capítulo 5", "Capítulo 6")
)

DT::datatable(equipamentos,
 options = list(pageLength = 10, dom = 'tip'),
 caption = "Equipamentos disponíveis no FABLAB UFPB")
```

## Plain Text

```
1.6 Capítulo de Impressoras 3D (02-impressoras-3d.Rmd)

```markdown
# Impressoras 3D {#impressoras-3d}

## Visão Geral

O FABLAB UFPB possui quatro impressoras 3D com diferentes tecnologias:

- **FDM (Fused Deposition Modeling)**: Prusa i3 MK2, MakerBot Replicator 2X,
  Ender 5 Plus
- **SLA (Stereolithography)**: XYZprinting Nobel 1.0A

## Ender 5 Plus {#ender-5-plus}

### Especificações Técnicas

```{r ender5-specs}
specs_ender5 <- data.frame(
 Especificação = c("Volume de Impressão", "Resolução de Camada",
 "Velocidade",
 "Temperatura Extrusor", "Temperatura Mesa",
 "Filamentos"),
 Valor = c("220×220×300 mm", "0.1-0.3 mm", "180 mm/s",
 "260°C máx", "100°C máx", "PLA, ABS, PETG, TPU")
)
```

```
)
```

```
knitr::kable(specs_ender5, caption = "Especificações Ender 5 Plus")
```

## Procedimento de Operação

### 1. Preparação da Impressora

Plain Text

```
Checklist pré-impressão
checklist_pre_impressao <- function() {
 cat("✓ Mesa nivelada\n")
 cat("✓ Filamento carregado\n")
 cat("✓ Bico limpo\n")
 cat("✓ Arquivo G-code preparado\n")
}
```

1. **Ligar a impressora** e aguardar inicialização
2. **Verificar nivelamento** da mesa
3. **Carregar filamento** no extrusor
4. **Pré-aquecer** mesa e bico conforme material

### 2. Configurações por Material

Plain Text

```
materiais <- data.frame(
 Material = c("PLA", "ABS", "PETG", "TPU"),
 Temp_Bico = c("200-220°C", "230-250°C", "220-250°C", "210-230°C"),
 Temp_Mesa = c("50-60°C", "80-100°C", "70-80°C", "40-60°C"),
 Velocidade = c("60 mm/s", "40 mm/s", "50 mm/s", "30 mm/s"),
 Observacoes = c("Fácil impressão", "Requer ventilação", "Resistente",
"Flexível")
)

DT::datatable(materiais,
 options = list(pageLength = 5, dom = 't'),
 caption = "Configurações por tipo de material")
```

## Manutenção Preventiva

## Cronograma de Manutenção

Plain Text

```
library(plotly)

cronograma <- data.frame(
 Atividade = c("Limpeza bico", "Lubrificação eixos", "Calibração mesa",
 "Troca filamento guia", "Revisão geral"),
 Frequencia = c("Semanal", "Mensal", "Quinzenal", "Trimestral",
 "Semestral"),
 Tempo_min = c(15, 30, 20, 45, 120),
 Prioridade = c("Alta", "Média", "Alta", "Baixa", "Alta")
)

p <- plot_ly(cronograma, x = ~Atividade, y = ~Tempo_min,
 color = ~Prioridade, type = 'bar',
 text = ~paste("Frequência:", Frequencia),
 textposition = 'outside') %>%
 layout(title = "Cronograma de Manutenção - Ender 5 Plus",
 xaxis = list(title = "Atividade"),
 yaxis = list(title = "Tempo (minutos)"))
```

p

## Solução de Problemas

### Problemas Comuns

Plain Text

```
problemas <- data.frame(
 Problema = c("Primeira camada não adere", "Filamento não sai", "Impressão
 com falhas",
 "Ruído excessivo", "Aquecimento irregular"),
 Causa_Provavel = c("Mesa desnivelada", "Bico entupido", "Configuração
 incorreta",
 "Correias frouxas", "Termistor defeituoso"),
 Solucao = c("Nivelar mesa", "Limpar/trocar bico", "Revisar slicer",
 "Ajustar correias", "Verificar conexões"),
 Urgencia = c("Alta", "Média", "Média", "Baixa", "Alta")
)

DT::datatable(problemas,
```

```
options = list(pageLength = 10, dom = 'tip'),
caption = "Guia de solução de problemas")
```

## Prusa i3 MK2 {#prusa-i3}

### Características Especiais

A Prusa i3 MK2 é conhecida por sua confiabilidade e qualidade de impressão.

Plain Text

```
Gráfico de comparação de qualidade
qualidade_dados <- data.frame(
 Aspecto = c("Precisão", "Velocidade", "Facilidade", "Confiabilidade"),
 Prusa_i3 = c(9, 7, 8, 9),
 Ender_5 = c(8, 8, 7, 8),
 MakerBot = c(7, 6, 9, 7)
)

p2 <- plot_ly(qualidade_dados, x = ~Aspecto, y = ~Prusa_i3,
 type = 'scatter', mode = 'lines+markers', name = 'Prusa i3
MK2') %>%
 add_trace(y = ~Ender_5, name = 'Ender 5 Plus') %>%
 add_trace(y = ~MakerBot, name = 'MakerBot 2X') %>%
 layout(title = "Comparação de Performance das Impressoras",
 yaxis = list(title = "Pontuação (1-10)"))
```

p2

Plain Text

```
1.7 Estilos Personalizados (style.css)

```css
/* Estilos personalizados para o manual FABLAB */

/* Cores do FABLAB */
:root {
  --fablab-green: #8BC34A;
  --fablab-blue: #4FC3F7;
  --fablab-gray: #757575;
}

/* Header personalizado */
```



```
.book .book-header {
  background: linear-gradient(135deg, var(--fablab-green), var(--fablab-blue));
  color: white;
}

/* Sidebar */
.book .book-summary {
  background: #f8f9fa;
  border-right: 2px solid var(--fablab-green);
}

/* Links ativos */
.book .book-summary ul.summary li.active > a {
  color: var(--fablab-green);
  font-weight: bold;
}

/* Tabelas */
.dataTables_wrapper {
  margin: 20px 0;
}

/* Botões */
.btn-primary {
  background-color: var(--fablab-green);
  border-color: var(--fablab-green);
}

/* Alertas personalizados */
.alert-info {
  border-left: 4px solid var(--fablab-blue);
}

.alert-warning {
  border-left: 4px solid #FFD54F;
}

.alert-danger {
  border-left: 4px solid #FF6B6B;
}

/* Código */
pre {
  background: #f8f9fa;
  border-left: 4px solid var(--fablab-green);
  padding: 15px;
}
```

```
/* Responsividade */
@media (max-width: 768px) {
  .book .book-body .page-wrapper .page-inner {
    padding: 10px;
  }
}
```

1.8 Comandos para Compilar

Plain Text

```
# No R Studio, execute:

# 1. Compilar o livro
bookdown::render_book("index.Rmd", "bookdown::gitbook")

# 2. Compilar PDF
bookdown::render_book("index.Rmd", "bookdown::pdf_book")

# 3. Compilar EPUB
bookdown::render_book("index.Rmd", "bookdown::epub_book")

# 4. Compilar todos os formatos
bookdown::render_book("index.Rmd", "all")

# 5. Servir localmente para desenvolvimento
bookdown::serve_book()
```



ABORDAGEM 2: FLEXDASHBOARD

Por que flexdashboard?

- ☒ Dashboard interativo
- ☒ Múltiplas páginas
- ☒ Widgets interativos
- ☒ Integração com Shiny
- ☒ Layout responsivo

2.1 Estrutura do Dashboard

Plain Text

```

---
title: "FABLAB UFPB - Manual Interativo"
output:
  flexdashboard::flex_dashboard:
    orientation: columns
    vertical_layout: fill
    theme: bootstrap
    navbar:
      - { title: "Home", href: "#home", align: left }
      - { title: "Impressoras 3D", href: "#impressoras", align: left }
      - { title: "Laser", href: "#laser", align: left }
      - { title: "CNC", href: "#cnc", align: left }
      - { title: "Manutenção", href: "#manutencao", align: left }
---

```{r setup, include=FALSE}
library(flexdashboard)
library(DT)
library(plotly)
library(shiny)

```

# Home {#home}

## Column 1

### Equipamentos do FABLAB

Plain Text

```

Tabela interativa de equipamentos
equipamentos <- data.frame(
 Equipamento = c("STORM 1390", "Ender 5 Plus", "Prusa i3 MK2", "Router
CNC"),
 Status = c("Operacional", "Operacional", "Manutenção", "Operacional"),
 Uso_Semanal = c(25, 40, 15, 20),
 Proxima_Manutencao = c("15/03/2024", "22/03/2024", "18/03/2024",
"30/03/2024")
)

DT::datatable(equipamentos, options = list(pageLength = 10))

```

## Column 2

### Status em Tempo Real

Plain Text

```
valueBox(value = "8",
 caption = "Equipamentos Ativos",
 icon = "fa-cogs",
 color = "success")
```

Plain Text

```
valueBox(value = "95%",
 caption = "Disponibilidade",
 icon = "fa-check-circle",
 color = "primary")
```

Plain Text

```
valueBox(value = "3",
 caption = "Manutenções Hoje",
 icon = "fa-wrench",
 color = "warning")
```

### Uso Semanal

Plain Text

```
uso_dados <- data.frame(
 Dia = c("Seg", "Ter", "Qua", "Qui", "Sex"),
 Horas = c(8, 12, 10, 15, 6)
)

plot_ly(uso_dados, x = ~Dia, y = ~Horas, type = 'bar') %>%
 layout(title = "Horas de Uso por Dia")
```

## Impressoras 3D {#impressoras}

### Column 1 {.tabset}

## Ender 5 Plus

### Especificações:

- Volume: 220×220×300 mm
- Resolução: 0.1-0.3 mm
- Materiais: PLA, ABS, PETG, TPU

### Status Atual:

Plain Text

```
gauge(85, min = 0, max = 100, symbol = '%',
 gaugeSectors(success = c(80, 100), warning = c(40, 79), danger = c(0,
39)))
```

## Prusa i3 MK2

### Especificações:

- Volume: 250×210×200 mm
- Auto-nivelamento: Sim
- Materiais: PLA, ABS, PETG

### Status Atual:

Plain Text

```
gauge(92, min = 0, max = 100, symbol = '%',
 gaugeSectors(success = c(80, 100), warning = c(40, 79), danger = c(0,
39)))
```

## Column 2

### Configurações por Material

Plain Text

```
materiais <- data.frame(
 Material = c("PLA", "ABS", "PETG", "TPU"),
 Temp_Bico = c(210, 240, 235, 220),
 Temp_Mesa = c(55, 90, 75, 50),
 Velocidade = c(60, 40, 50, 30)
```

```
)

plot_ly(materiais, x = ~Material, y = ~Temp_Bico,
 type = 'bar', name = 'Temperatura Bico') %>%
 add_trace(y = ~Temp_Mesa, name = 'Temperatura Mesa') %>%
 layout(title = "Configurações por Material")
```

# Manutenção {#manutencao}

## Column 1

### Cronograma de Manutenção

Plain Text

```
cronograma <- data.frame(
 Equipamento = c("Ender 5", "Prusa i3", "STORM 1390", "Router CNC"),
 Proxima = c("22/03", "18/03", "15/03", "30/03"),
 Tipo = c("Preventiva", "Corretiva", "Preventiva", "Preventiva"),
 Prioridade = c("Média", "Alta", "Baixa", "Média")
)

DT::datatable(cronograma, options = list(pageLength = 10))
```

## Column 2

### Custos de Manutenção

Plain Text

```
custos <- data.frame(
 Mes = c("Jan", "Fev", "Mar", "Abr", "Mai"),
 Preventiva = c(500, 300, 400, 350, 450),
 Corretiva = c(200, 800, 150, 600, 100)
)

plot_ly(custos, x = ~Mes, y = ~Preventiva,
 type = 'scatter', mode = 'lines+markers', name = 'Preventiva') %>%
 add_trace(y = ~Corretiva, name = 'Corretiva') %>%
 layout(title = "Custos de Manutenção (R$)")
```

## Plain Text

---

## 🚀 ABORDAGEM 3: SHINY APP

### \*\*Por que Shiny?\*\*

- ✅ Aplicação web completa
- ✅ Interatividade total
- ✅ Atualizações em tempo real
- ✅ Integração com banco de dados
- ✅ Controle de usuários

### \*\*3.1 Estrutura da Aplicação Shiny\*\*

```r

app.R - Aplicação Shiny completa

library(shiny)

library(shinydashboard)

library(DT)

library(plotly)

library(shinyWidgets)

UI

ui <- dashboardPage(

dashboardHeader(title = "FABLAB UFPB - Manual Interativo"),

dashboardSidebar(

sidebarMenu(

menuItem("Dashboard", tabName = "dashboard", icon = icon("dashboard")),

menuItem("Impressoras 3D", tabName = "impressoras", icon =

icon("cube")),

menuItem("Corte a Laser", tabName = "laser", icon = icon("cut")),

menuItem("Router CNC", tabName = "cnc", icon = icon("cogs")),

menuItem("Manutenção", tabName = "manutencao", icon = icon("wrench")),

menuItem("Relatórios", tabName = "relatorios", icon = icon("chart-

bar"))

)

),

dashboardBody(

tags\$head(

tags\$style(HTML("

.main-header .navbar {

background: linear-gradient(135deg, #8BC34A, #4FC3F7) !important;

}

```

        .skin-blue .main-sidebar {
            background-color: #f8f9fa;
        }
    ))
),

tabItems(
    # Dashboard principal
    tabItem(tabName = "dashboard",
        fluidRow(
            valueBoxOutput("total_equipamentos"),
            valueBoxOutput("disponibilidade"),
            valueBoxOutput("manutencoes_hoje")
        ),

        fluidRow(
            box(title = "Status dos Equipamentos", status = "primary",
                solidHeader = TRUE, width = 6,
                DT::dataTableOutput("tabela_equipamentos")),

            box(title = "Uso Semanal", status = "success",
                solidHeader = TRUE, width = 6,
                plotlyOutput("grafico_uso"))
        )
    ),

    # Impressoras 3D
    tabItem(tabName = "impressoras",
        fluidRow(
            box(title = "Seleção de Impressora", status = "primary",
                solidHeader = TRUE, width = 3,
                selectInput("impressora_selecionada", "Escolha a impressora:",
                    choices = c("Ender 5 Plus", "Prusa i3 MK2",
"MakerBot 2X")),

            hr(),

            h4("Configurações Rápidas"),
            selectInput("material", "Material:",
                choices = c("PLA", "ABS", "PETG", "TPU")),

            actionButton("aplicar_config", "Aplicar Configuração",
                class = "btn-success")
        ),

        box(title = "Especificações", status = "info",
            solidHeader = TRUE, width = 9,
            DT::dataTableOutput("specs_impressora"))
    )
)

```



```

    ),

    fluidRow(
      box(title = "Configurações por Material", status = "warning",
        solidHeader = TRUE, width = 6,
        plotlyOutput("config_materiais")),

      box(title = "Histórico de Uso", status = "success",
        solidHeader = TRUE, width = 6,
        plotlyOutput("historico_uso"))
    )
  ),

  # Manutenção
  tabItem(tabName = "manutencao",
    fluidRow(
      box(title = "Agendar Manutenção", status = "primary",
        solidHeader = TRUE, width = 4,
        selectInput("equip_manutencao", "Equipamento:",
          choices = c("Ender 5 Plus", "STORM 1390", "Router
CNC"))),

      dateInput("data_manutencao", "Data:", value = Sys.Date()),

      selectInput("tipo_manutencao", "Tipo:",
        choices = c("Preventiva", "Corretiva",
"Calibração")),

      textAreaInput("observacoes", "Observações:", rows = 3),

      actionButton("agendar", "Agendar Manutenção",
        class = "btn-primary")
    ),

    box(title = "Cronograma", status = "success",
      solidHeader = TRUE, width = 8,
      DT::dataTableOutput("cronograma_manutencao"))
  ),

  fluidRow(
    box(title = "Custos de Manutenção", status = "warning",
      solidHeader = TRUE, width = 12,
      plotlyOutput("custos_manutencao"))
  )
)
)
)
)
)

```

```

# Server
server <- function(input, output, session) {

  # Dados reativos
  equipamentos_data <- reactive({
    data.frame(
      Equipamento = c("STORM 1390", "Ender 5 Plus", "Prusa i3 MK2", "Router
CNC"),
      Status = c("Operacional", "Operacional", "Manutenção", "Operacional"),
      Uso_Horas = c(120, 200, 80, 150),
      Disponibilidade = c("98%", "95%", "85%", "92%")
    )
  })

  # Value boxes
  output$total_equipamentos <- renderValueBox({
    valueBox(
      value = 8,
      subtitle = "Total de Equipamentos",
      icon = icon("cogs"),
      color = "green"
    )
  })

  output$disponibilidade <- renderValueBox({
    valueBox(
      value = "95%",
      subtitle = "Disponibilidade Média",
      icon = icon("check-circle"),
      color = "blue"
    )
  })

  output$manutencoes_hoje <- renderValueBox({
    valueBox(
      value = 3,
      subtitle = "Manutenções Hoje",
      icon = icon("wrench"),
      color = "yellow"
    )
  })

  # Tabelas
  output$tabela_equipamentos <- DT::renderDataTable({
    DT::datatable(equipamentos_data(), options = list(pageLength = 10))
  })
}

```

```

# Gráficos
output$grafico_uso <- renderPlotly({
  uso_dados <- data.frame(
    Dia = c("Seg", "Ter", "Qua", "Qui", "Sex"),
    Horas = c(8, 12, 10, 15, 6)
  )

  plot_ly(uso_dados, x = ~Dia, y = ~Horas, type = 'bar',
    marker = list(color = '#8BC34A')) %>%
    layout(title = "Uso Semanal (Horas)")
})

# Especificações da impressora selecionada
output$specs_imprensa <- DT::renderDataTable({
  specs <- switch(input$impressora_selecionada,
    "Ender 5 Plus" = data.frame(
      Especificacao = c("Volume", "Resolução", "Velocidade", "Materiais"),
      Valor = c("220×220×300 mm", "0.1-0.3 mm", "180 mm/s", "PLA, ABS,
PETG, TPU")
    ),
    "Prusa i3 MK2" = data.frame(
      Especificacao = c("Volume", "Resolução", "Auto-nivelamento",
"Materiais"),
      Valor = c("250×210×200 mm", "0.05-0.35 mm", "Sim", "PLA, ABS, PETG")
    ),
    data.frame(
      Especificacao = c("Volume", "Resolução", "Extrusores", "Materiais"),
      Valor = c("246×152×155 mm", "0.1-0.34 mm", "Duplo", "PLA, ABS")
    )
  )

  DT::datatable(specs, options = list(dom = 't'))
})

# Observador para aplicar configurações
observeEvent(input$aplicar_config, {
  showNotification(
    paste("Configuração aplicada para", input$material, "na",
input$impressora_selecionada),
    type = "success"
  )
})
}

# Executar aplicação
shinyApp(ui = ui, server = server)

```

IMPLEMENTAÇÃO PRÁTICA

Passo a Passo Completo

1. Preparação do Ambiente

Plain Text

```
# 1. Instalar R e RStudio
# Download: https://www.r-project.org/ e https://www.rstudio.com/

# 2. Instalar pacotes necessários
install.packages(c(
  "bookdown", "rmarkdown", "knitr", "DT", "plotly",
  "flexdashboard", "shiny", "shinydashboard", "shinyWidgets",
  "htmlwidgets", "crosstalk", "leaflet", "dygraphs"
))

# 3. Criar projeto no RStudio
# File > New Project > New Directory > Book Project using bookdown
```

2. Estruturação dos Dados

Plain Text

```
# Criar base de dados dos manuais
criar_base_dados <- function() {

  # Equipamentos
  equipamentos <- data.frame(
    id = 1:8,
    nome = c("STORM 1390", "Prusa i3 MK2", "MakerBot Replicator 2X",
             "XYZprinting Nobel 1.0A", "Ender 5 Plus", "Router CNC",
             "Plotter VISUTECH", "Bungard CCD/2/ATC"),
    categoria = c("Laser", "Impressão 3D", "Impressão 3D", "Impressão 3D",
                  "Impressão 3D", "CNC", "Corte", "PCB"),
    status = c("Operacional", "Operacional", "Manutenção", "Operacional",
               "Operacional", "Operacional", "Operacional", "Operacional")
  )

  # Procedimentos
  procedimentos <- data.frame(
    equipamento_id = c(1, 1, 2, 2, 5, 5),
    tipo = c("Operação", "Manutenção", "Operação", "Manutenção", "Operação",
             "Manutenção"),
```

```

    titulo = c("Configuração de Corte", "Limpeza das Lentes",
               "Preparação de Impressão", "Calibração da Mesa",
               "Carregamento de Filamento", "Lubrificação dos Eixos"),
    descricao = c("Procedimento para configurar parâmetros de corte...",
                  "Limpeza periódica das lentes do laser...",
                  "Preparação da impressora para nova impressão...",
                  "Calibração manual da mesa de impressão...",
                  "Carregamento correto do filamento...",
                  "Lubrificação dos eixos X, Y e Z...")
  )

# Salvar dados
save(equipamentos, procedimentos, file = "dados_fablab.RData")

return(list(equipamentos = equipamentos, procedimentos = procedimentos))
}

# Executar
dados <- criar_base_dados()

```

3. Desenvolvimento Iterativo

Plain Text

```

# Workflow de desenvolvimento

# 1. Desenvolvimento local
bookdown::serve_book() # Servidor local para testes

# 2. Compilação para produção
bookdown::render_book("index.Rmd", "bookdown::gitbook")

# 3. Deploy (opções)
# - GitHub Pages
# - Netlify
# - Servidor próprio
# - RStudio Connect

```

Funcionalidades Avançadas

1. Busca Inteligente

Plain Text

```
# Implementar busca personalizada
search_manual <- function(termo, dados) {
  resultados <- dados[grepl(termo, dados$conteudo, ignore.case = TRUE), ]
  return(resultados)
}
```

2. Integração com Banco de Dados

Plain Text

```
# Conectar com banco de dados real
library(DBI)
library(RSQLite)

# Criar conexão
con <- dbConnect(SQLite(), "fablab_manual.db")

# Criar tabelas
dbExecute(con, "
  CREATE TABLE IF NOT EXISTS equipamentos (
    id INTEGER PRIMARY KEY,
    nome TEXT,
    categoria TEXT,
    status TEXT,
    data_atualizacao DATE
  )
")

# Inserir dados
dbWriteTable(con, "equipamentos", equipamentos, append = TRUE)
```

3. Relatórios Automáticos

Plain Text

```
# Gerar relatórios automáticos
gerar_relatorio_manutencao <- function() {
  rmarkdown::render(
    "relatorio_manutencao.Rmd",
    output_file = paste0("relatorio_", Sys.Date(), ".html"),
    params = list(data_inicio = Sys.Date() - 30, data_fim = Sys.Date())
  )
}
```

DEPLOY E PUBLICAÇÃO

Opções de Hospedagem

1. GitHub Pages (Gratuito)

Bash

```
# 1. Criar repositório no GitHub
# 2. Fazer push do projeto
git add .
git commit -m "Manual FABLAB inicial"
git push origin main

# 3. Ativar GitHub Pages nas configurações
# 4. Acessar via: https://usuario.github.io/manual-fablab
```

2. Netlify (Gratuito)

Plain Text

```
# 1. Compilar o livro
bookdown::render_book("index.Rmd", "bookdown::gitbook")

# 2. Fazer upload da pasta _book/ no Netlify
# 3. Configurar domínio personalizado (opcional)
```

3. Shinyapps.io (Para Shiny)

Plain Text

```
# 1. Instalar rsconnect
install.packages("rsconnect")

# 2. Configurar conta
rsconnect::setAccountInfo(name='<ACCOUNT>',
                           token='<TOKEN>',
                           secret='<SECRET>')

# 3. Deploy
rsconnect::deployApp()
```



EXEMPLO PRÁTICO COMPLETO

Manual da Ender 5 Plus - Versão Completa

Plain Text

```
# ender5_manual.Rmd

---
title: "Manual Ender 5 Plus"
output:
  bookdown::html_document2:
    toc: true
    toc_float: true
    theme: flatly
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = FALSE, message = FALSE, warning = FALSE)
library(DT)
library(plotly)
library(knitr)
```

## Ender 5 Plus - Manual Completo

### Especificações Técnicas

Plain Text

```
specs <- data.frame(
 Característica = c("Volume de Impressão", "Resolução de Camada",
 "Velocidade Máxima",
 "Temperatura Extrusor", "Temperatura Mesa",
 "Conectividade",
 "Filamentos Suportados", "Precisão de Posicionamento"),
 Especificação = c("220 × 220 × 300 mm", "0.1 - 0.3 mm", "180 mm/s",
 "260°C", "100°C", "USB, SD Card, WiFi",
 "PLA, ABS, PETG, TPU, Wood, Metal", "±0.1 mm")
)

knitr::kable(specs, caption = "Especificações Técnicas da Ender 5 Plus")
```



# Configurações por Material

Plain Text

```
materiais <- data.frame(
 Material = c("PLA", "ABS", "PETG", "TPU", "Wood Fill", "Metal Fill"),
 Temp_Bico = c("200-220°C", "230-250°C", "220-250°C", "210-230°C", "190-220°C", "190-220°C"),
 Temp_Mesa = c("50-60°C", "80-100°C", "70-80°C", "40-60°C", "45-60°C", "45-60°C"),
 Velocidade = c("60 mm/s", "40 mm/s", "50 mm/s", "30 mm/s", "40 mm/s", "40 mm/s"),
 Retração = c("6 mm", "6 mm", "6 mm", "2 mm", "6 mm", "6 mm"),
 Observações = c("Fácil impressão", "Ventilação necessária", "Resistente químico",
 "Material flexível", "Acabamento madeira", "Acabamento metálico")
)

DT::datatable(materiais,
 options = list(pageLength = 10, scrollX = TRUE),
 caption = "Configurações recomendadas por tipo de material")
```

## Procedimentos de Operação

### Preparação da Impressora

#### 1. Verificação Inicial

- ☒ Verificar se a mesa está limpa
- ☒ Conferir se não há filamento antigo no extrusor
- ☒ Verificar se os eixos estão livres

#### 2. Nivelamento da Mesa

Plain Text

```
nivelamento <- data.frame(
 Passo = 1:5,
 Ação = c("Aquecer mesa para 60°C", "Posicionar bico no canto frontal esquerdo",
 "Ajustar altura com papel sulfite", "Repetir nos 4 cantos", "Verificar centro da mesa"),
 Tempo_Estimado = c("3 min", "1 min", "2 min", "5 min", "1 min")
)
```

```
knitr::kable(nivelamento, caption = "Procedimento de Nivelamento da Mesa")
```

## Carregamento de Filamento

Plain Text

```
Função para calcular quantidade de filamento necessária
calcular_filamento <- function(peso_objeto_g, densidade_material = 1.24) {
 # PLA densidade média: 1.24 g/cm³
 volume_cm3 <- peso_objeto_g / densidade_material
 comprimento_m <- volume_cm3 / (pi * (1.75/2)^2) / 100 # Filamento 1.75mm
 return(round(comprimento_m, 2))
}

Exemplo: objeto de 50g
cat("Filamento necessário:", calcular_filamento(50), "metros")
```

## Manutenção Preventiva

### Cronograma de Manutenção

Plain Text

```
cronograma <- data.frame(
 Frequência = c("Diária", "Semanal", "Quinzenal", "Mensal", "Trimestral",
"Semestral"),
 Atividades = c("Verificação visual geral",
 "Limpeza da mesa e bico",
 "Verificação de correias",
 "Lubrificação dos eixos",
 "Calibração completa",
 "Revisão geral e troca de peças"),
 Tempo_min = c(2, 15, 10, 30, 60, 120),
 Responsável = c("Usuário", "Usuário", "Técnico", "Técnico", "Técnico",
"Especialista")
)

Gráfico de tempo de manutenção
p <- plot_ly(cronograma, x = ~Frequência, y = ~Tempo_min,
 type = 'bar', marker = list(color = '#8BC34A')) %>%
 layout(title = "Tempo de Manutenção por Frequência",
 xaxis = list(title = "Frequência"),
 yaxis = list(title = "Tempo (minutos)"))
```

## Checklist de Manutenção Semanal

Plain Text

```
checklist_semanal <- data.frame(
 Item = c("Mesa de impressão limpa", "Bico desobstruído", "Filamento sem
emaranhados",
 "Correias tensionadas", "Eixos lubrificados", "Ventoinhas
funcionando",
 "Cabos conectados", "Software atualizado"),
 Status = c("✅", "✅", "⚠️", "✅", "✅", "✅", "✅", "❌"),
 Observações = c("OK", "OK", "Verificar rolo", "OK", "OK", "OK", "OK",
"Atualizar firmware")
)

DT::datatable(checklist_semanal,
 options = list(dom = 't'),
 caption = "Checklist de Manutenção Semanal - Exemplo")
```

## Solução de Problemas

### Problemas Comuns e Soluções

Plain Text

```
problemas <- data.frame(
 Problema = c("Primeira camada não adere", "Filamento não sai do bico",
 "Impressão com camadas desalinhadas", "Ruído excessivo
durante impressão",
 "Aquecimento irregular da mesa", "Falha na extrusão"),
 Causa_Provável = c("Mesa desnivelada ou suja", "Bico entupido ou
temperatura baixa",
 "Correias frouxas ou eixos sujos", "Correias muito
tensas ou eixos secos",
 "Termistor defeituoso ou conexão solta", "Filamento
emaranhado ou extrusor com problema"),
 Solução = c("Nivelar mesa e limpar com álcool", "Limpar bico ou aumentar
temperatura",
 "Ajustar correias e lubrificar eixos", "Reduzir tensão das
correias",
 "Verificar conexões do termistor", "Verificar caminho do
filamento"),
```

```

 Urgência = c("Alta", "Média", "Média", "Baixa", "Alta", "Média"),
 Tempo_Solução = c("10 min", "20 min", "15 min", "5 min", "30 min", "15
min")
)

DT::datatable(problemas,
 options = list(pageLength = 10, scrollX = TRUE),
 caption = "Guia de Solução de Problemas")

```

## Códigos de Erro

Plain Text

```

codigos_erro <- data.frame(
 Código = c("E1", "E2", "E3", "E4", "E5"),
 Descrição = c("Erro de aquecimento do extrusor", "Erro de aquecimento da
mesa",
 "Erro de termistor", "Erro de movimento", "Erro de
comunicação"),
 Ação = c("Verificar resistência do extrusor", "Verificar resistência da
mesa",
 "Verificar conexão do termistor", "Verificar motores e correias",
 "Verificar cabo USB/SD")
)

knitr::kable(codigos_erro, caption = "Códigos de Erro e Ações Corretivas")

```

## Histórico de Uso

Plain Text

```

Simular dados de uso
set.seed(123)
datas <- seq(from = as.Date("2024-01-01"), to = Sys.Date(), by = "day")
horas_uso <- round(runif(length(datas), 0, 8), 1)

uso_historico <- data.frame(
 Data = datas,
 Horas_Uso = horas_uso,
 Projetos = sample(1:5, length(datas), replace = TRUE)
)

Gráfico de uso ao longo do tempo
p2 <- plot_ly(uso_historico, x = ~Data, y = ~Horas_Uso,
 type = 'scatter', mode = 'lines',

```

```

 line = list(color = '#4FC3F7')) %>%
 layout(title = "Histórico de Uso da Ender 5 Plus",
 xaxis = list(title = "Data"),
 yaxis = list(title = "Horas de Uso"))

```

p2

## Relatório de Performance

Plain Text

```

Métricas de performance
metricas <- data.frame(
 Métrica = c("Tempo Médio de Impressão", "Taxa de Sucesso", "Consumo Médio
de Filamento",
 "Tempo Médio de Preparação", "Disponibilidade", "Custo por
Hora"),
 Valor = c("4.5 horas", "94%", "85g", "15 min", "96%", "R$ 2.50"),
 Meta = c("< 5 horas", "> 90%", "< 100g", "< 20 min", "> 95%", "< R$ 3.00"),
 Status = c("✅", "✅", "✅", "✅", "✅", "✅")
)

knitr::kable(metricas, caption = "Relatório de Performance - Último Mês")

```

Plain Text

---

## 🎯 CONCLUSÃO E PRÓXIMOS PASSOS

### \*\*Recomendação Final\*\*

Para o FABLAB UFPB, recomendo a **abordagem bookdown** porque:

1. ✅ **Profissional**: Ideal para documentação técnica
2. ✅ **Escalável**: Fácil de expandir e manter
3. ✅ **Multiplataforma**: HTML, PDF, EPUB
4. ✅ **Busca Integrada**: Encontrar informações rapidamente
5. ✅ **Colaborativo**: Múltiplos autores podem contribuir






### **Cronograma de Implementação**

```r






```
cronograma_implementacao <- data.frame(
```

```
Semana = 1:8,  
Atividade = c("Setup do ambiente R", "Estruturação do projeto",  
              "Criação dos capítulos principais", "Desenvolvimento de  
conteúdo",  
              "Implementação de interatividade", "Testes e revisões",  
              "Deploy e configuração", "Treinamento da equipe"),  
Responsável = c("Técnico", "Equipe", "Equipe", "Especialistas",  
                "Técnico", "Todos", "Técnico", "Coordenador")  
)
```

Recursos Necessários

-  **R Studio** (gratuito)
-  **Conhecimento básico de R Markdown**
-  **Hospedagem web** (GitHub Pages gratuito)
-  **40-60 horas** de desenvolvimento inicial
-  **2-3 pessoas** na equipe de desenvolvimento

Benefícios Esperados

-  **Acesso universal** aos manuais
-  **Busca rápida** de procedimentos
-  **Conteúdo interativo** e visual
-  **Atualizações fáceis** e versionamento
-  **Melhoria contínua** baseada no uso

O livro interativo em R será uma ferramenta poderosa para modernizar a documentação do FABLAB UFPB! 🚀